

AWS Lightsail에 Nodejs 배포하기

2024. 04. 13

OneITAcademy 학원

1.AWS Lightsail 계정만들고 서버 접속 설정하기

◆ AWS Lightsail 이란?

- 아마존에서 운영하는 웹 서비스에 특화된 클라우드 서비스
- 처음 3달은 무료, 그 이후부터 월 5달러 비용 청구됨 (비용을 내지 않으려면, 3달만 사용하고 삭제하면 됨)
-

◆ AWS 계정 생성

- <https://aws.amazon.com/ko> 를 접속한다
- [AWS 계정 생성] 버튼 누름
- 그 이후, 안내하는 절차에 따라, 계정 생성을 진행함
- 이름과 주소는 영문으로 입력해야 함. (juso.go.kr에서 '영문보기'를 참조 함)
- AWS 결제 정보에서 신용카드를 등록함
- [AWS에 가입] -> [Support 플랜 선택] 에서는 '기본지원 - 무료'를 선택한다
- 그러면, 가입이 완료됨

◆ AWS Lightsail 인스턴스 생성하기

- AWS Lightsail 홈페이지 접속 : <https://lightsail.aws.amazon.com> 하고, '루트 사용자'로 로그인 한다
- 처음 로그인하면, '안녕하세요!'라는 인사말과 함께, '지금은 인스턴스가 없습니다'라고 표시됨.
- '인스턴스 생성' 버튼을 누름
- 맨 위, [인스턴스의 위치]를 대한민국으로 변경
- [인스턴스 이미지 선택]에서, '플랫폼 선택'은 Linux/Unix를 선택하고, '블루프린터 선택' -> '운영 체제(OS) 전용' 선택 후, 'Ubuntu' 선택
- [인스턴스 플랜 선택]에서, '듀얼 스택'을 선택하고, '크기 선택'에서 '월별 5\$ 선택'
- [인스턴스 확인]에서 'Ubuntu-1' 입력된 그대로 두고,
- 아래쪽 '인스턴스 생성' 버튼 누름. ('대기중'에서 '실행중'이 나올 때까지 기다림)
- 그러면 바탕화면에 새롭게 생성된 인스턴스가 나타남.

◆ 고정 IP 생성하기

- 왼쪽 메뉴에서 [네트워킹] 선택하고, 화면에서 '고정IP 생성' 버튼 누름
- '고정 IP 위치'에서 '대한민국' 선택하고, '고정 IP 확인'에서 'Staticip-1'을 확인 후, '생성' 버튼 누름

◆ 방화벽 설정

- 왼쪽 메뉴에서 '인스턴스'를 선택하고, 화면에 나타난 인스턴스를 선택함.
- 화면의 네트워킹 탭을 선택하고, 'IPv4 방화벽'을 설정함. '+규칙추가' 를 눌러 다음과 같이 추가함

+ 규칙 추가

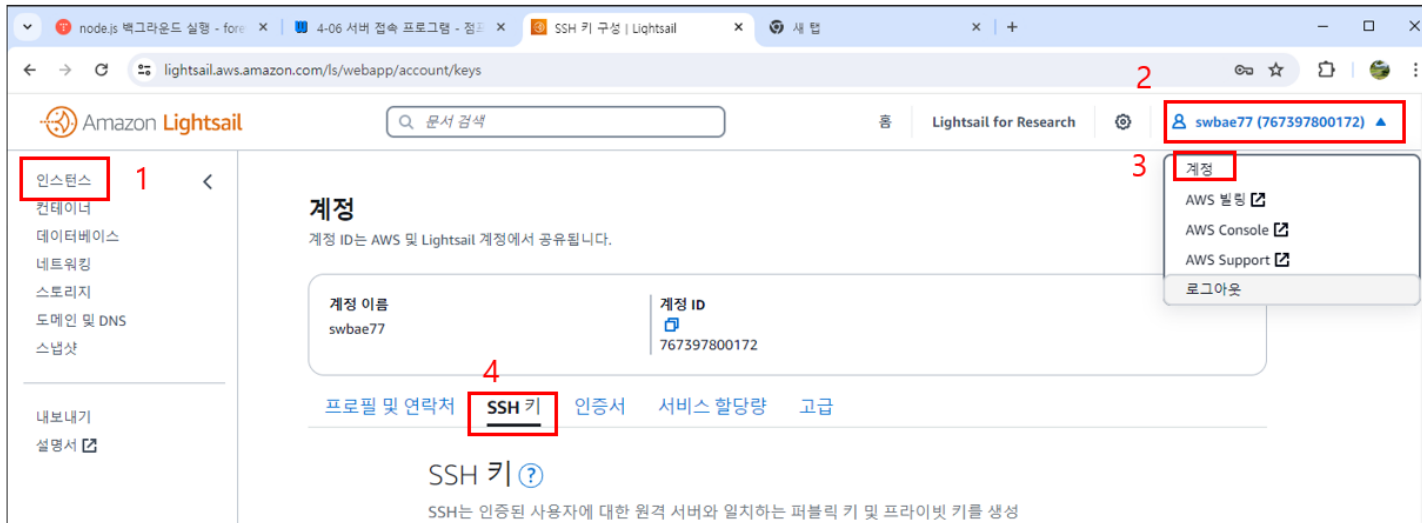
애플리케이션	프로토콜	포트 또는 범위 / 코드	제한:		
SSH	TCP	22	모든 IPv4 주소 Lightsail 브라우저 SSH/RDP ⓘ	☑	🛑
HTTP	TCP	80	모든 IPv4 주소	☑	🛑
HTTPS	TCP	443	모든 IPv4 주소	☑	🛑
사용자 지정	TCP	3500	모든 IPv4 주소	☑	🛑

443 포트 : https 설정을 위한 것임

3500 포트 : Nodejs에서의 서버 구동 포트 (배포하고자 하는 Nodejs의 서비스 포트)

2. 서버 접속 프로그램

◆ SSH 프라이빗 키 다운 받기



- 아래 다운받기를 눌러 다운로드 받은 후, C: 드라이브에 복사하여, 'mysite.pem'으로 이름 변경함.

+ 키 페어 생성

Region name ▼	Region code	Created	
🇰🇷 서울	ap-northeast-2	2024년 4월 11일 오후 2:32	📄 🗑
🇺🇸 버지니아	us-east-1	2024년 4월 12일 오후 5:08	📄 🗑

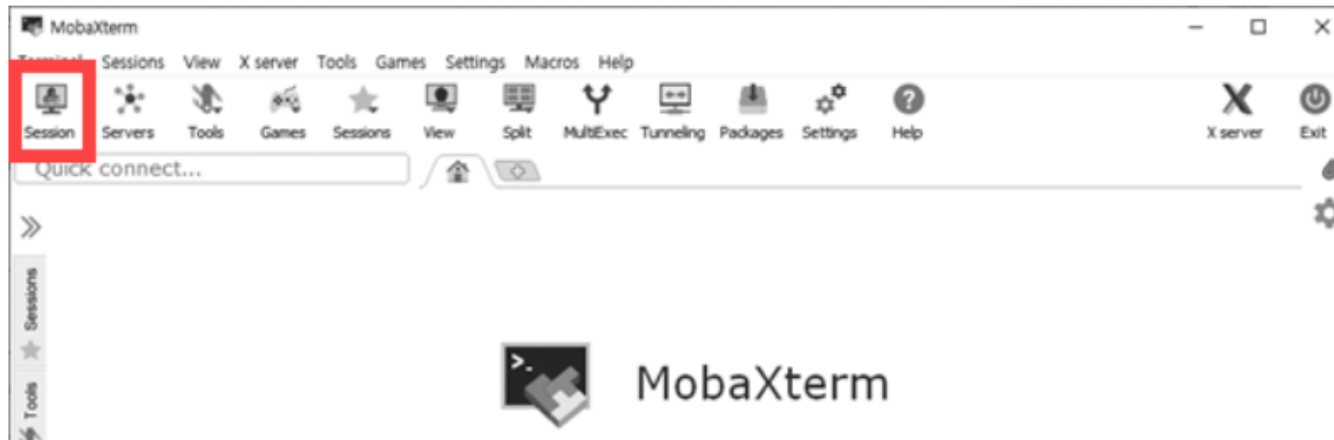
2개의 항목

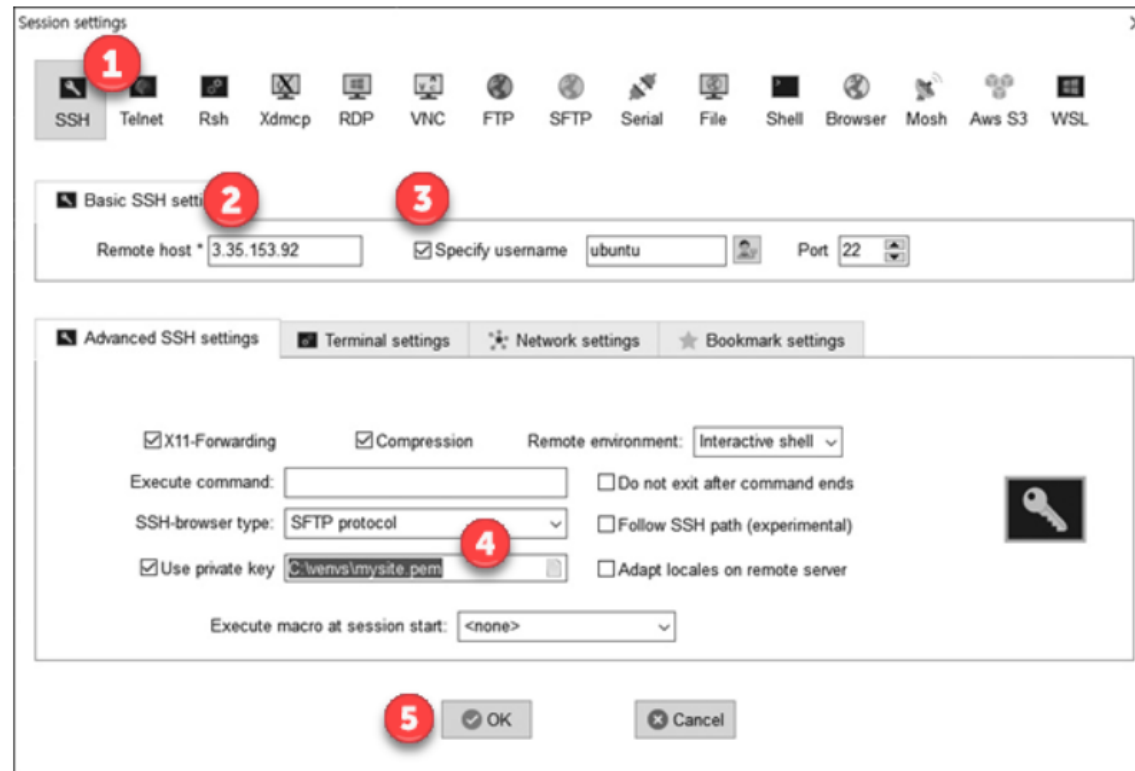
- Mac OS 인 경우, /users/<사용자명> 디렉토리에 복사함. 그런 다음 권한을 600 으로 변경

Chmod 600 mysite.pem

◆ MobaXterm 다운받고 설치하기

- MobaXterm 다운로드 주소 : mobaxterm.mobatek.net/download.html
- 설치가 완료되었으면, AWS Lightsail에 접속하기





1. SSH 누르기
2. 서버의 고정 IP 주소 입력(여러분의 고정 IP 주소를 입력해야 함)
3. 체크하고 **ubuntu** 입력
4. 체크하고 입력 창 오른쪽에 있는 서류 모양 아이콘을 눌러 로컬 컴퓨터에 있는 프라이빗 키 파일(mysite.pem) 선택
5. **<OK>** 누르기

◆ Mac OS 인 경우

- Mac 에서 기본적으로 지원하는 터미널 프로그램 이용하면 됨.

```
ssh -i ~/mysite.pem ubuntu@3.37.58.70
```


3. Nodejs 설치

AWS Lightsail 터미널에서 아래의 순서대로 설치함

```
sudo apt-get update
sudo apt-get install build-essential
sudo apt-get install curl
curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash --
sudo apt-get install nodejs
```

4. Source 설치 및 구동

- 구동할 NodeJS의 Source를 github에서 가져와서 컴파일 후 구동 확인함

```
git clone https://<github>레포지토리>
```

```
npm install
```

```
node server.js
```

확인사항) server.js에서 사용된 port가 Lightsail의 network에서 open되어 있는지 확인할 것

- VSC Thunder Client에서 Lightsail 서버로 접속확인함.

◆ MongoDB Atlas 설정

- 만약 MongoDB Atlas를 사용하고 있다면, AWS의 Lightsail 서버가 접근할 수 있도록 IP 주소 등록해줌.
- Project와 Database를 선택하고, 왼쪽 메뉴에서 Security > Network Access에서 'IP Access List' 탭의 'Add IP Address' 버튼을 눌러 서버의 IP 주소를 등록함.

Network Access

IP Access List

Peering

Private Endpoint

+ ADD IP ADDRESS

You will only be able to connect to your cluster from the following list of IP Addresses:

IP Address	Comment	Status	Actions
220.119.115.30/32 (includes your current IP address)	Created as part of the Auto Setup process	● Active	<button>⚙ EDIT</button> <button>🗑 DELETE</button>
43.200.15.180/32	ubuntu	● Active	<button>⚙ EDIT</button> <button>🗑 DELETE</button>
3.36.24.168/32	academy	● Active	<button>⚙ EDIT</button> <button>🗑 DELETE</button>

5.Nginx 설치

- Nginx 설치

```
sudo apt install nginx
```

◆ Nginx 설정

```
sudo rm /etc/nginx/sites-available/default
```

```
sudo rm /etc/nginx/sites-enabled/default
```

```
cd /etc/nginx/sites-available
```

```
sudo vi <projectname>
```

- 아래의 내용을 입력하고, 저장한다.

```
server {
```

```
    listen 80;
```

```
server_name 3.37.58.70;
```

```
location /static {
```

```
    alias /home/ubuntu/<projectsname>/public;
```

```
}
```

```
}
```

- 다음 생성한 파일을 링크한다.

```
cd /etc/nginx/sites-enabled
```

```
sudo ln -s /etc/nginx/sites-available/<projectname>
```

- Nginx 실행, 중지, 상태 확인

```
sudo systemctl start nginx
```

```
sudo systemctl stop nginx
```

```
sudo nginx -t
```

- 이제 http://<서버IP>로 접속할 수 있다. (포트 No. 생략 가능함)

6. 도메인 등록

- '내도메인.한국' 에서 무료 도메인 등록함.
- 적당한 도메인을 찾고, 상세정보를 입력하는 페이지에서 웹 서버 주소를 입력한다.

고급설정 (DNS)			
<input checked="" type="checkbox"/> IP연결(A)	<input type="text" value="test-go.r-e.kr"/>	<input type="text" value=".235"/>	[+] [-]
<input type="checkbox"/> IP연결(AAAA)	<input type="text" value="test-go.r-e.kr"/>	<input type="text" value="(예) 2001:0db8:85a3:08d3:1319:8a2e:0370:7334"/>	[+] [-]

- 도메인 이름으로 접속확인을 한다.

7.SSL 적용 (https)

- Client를 github에 배포하면, https로 서버에 접속하게 된다.
- 이때, 서버가 https를 적용하지 않으면 접속이 안된다. 따라서, 서버에 https가 적용되게 해야 한다.

◆ 인증서 설치

```
sudo apt install certbot
```

```
sudo apt install python3-certbot-nginx
```

- 그리고 아래의 명령어로 인증서를 생성한다. 이때, e-mail 주소를 입력하고, (A)gree를 위해 a를 입력하고, 앞서 생성한 도메인 이름을 입력한다. 예) mysite.com

```
sudo certbot certonly --nginx
```

- 인증서는 아래에 생성된다.

```
/etc/letsencrypt/live/<도메인네임>/fullchain.pem
```

```
/etc/letsencrypt/live/<도메인네임>/privkey.pem
```

◆ Nginx 설정

SSL 인증서를 Nginx에 적용하기 위해 다음과 같이 `/etc/nginx/sites-available/<project>` 파일을 수정해 주어야 한다

```
server {  
  
    listen 80;  
  
    server_name jovely12.kro.kr;  
  
    rewrite      ^ https://$server_name$request_uri? permanent;  
  
}
```

```
server {  
  
    listen 443;  
  
    server_name jovely12.kro.kr;  
  
  
    ssl on;  
  
    ssl_certificate /etc/letsencrypt/live/jovely12.kro.kr/fullchain.pem; # managed by Certbot  
    ssl_certificate_key /etc/letsencrypt/live/jovely12.kro.kr/privkey.pem; # managed by Certbot  
  
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3;  
  
    ssl_ciphers "ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384";
```

```
ssl_prefer_server_ciphers on;
```

```
location /static {
```

```
    alias /home/ubuntu/portfolio-server/public;
```

```
}
```

```
location / {
```

```
    proxy_pass http://localhost:3500;
```

```
    proxy_set_header X-Real-IP $remote_addr;
```

```
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
```

```
    proxy_set_header Host $http_host;
```

```
}
```

```
}
```

~

- Nginx를 다시 start 한다.

```
sudo systemctl stop nginx
```

```
sudo systemctl start nginx
```


- SSL 방화벽 설정을 다시 확인한다. 즉, Lightsail에서 443번 포트의 해제가 필요하다. (확인함)

IPv4 방화벽 ?

인터넷 또는 특정 IPv4 주소 또는 범위에 대해 포트를 개방하는 규칙을 생성합니다.

[방화벽 규칙에 대해 자세히 알아보기](#)

+ 규칙 추가

애플리케이션	프로토콜	포트 또는 범위 / 코드	제한:		
SSH	TCP	22	모든 IPv4 주소 Lightsail 브라우저 SSH/RDP ?		
HTTP	TCP	80	모든 IPv4 주소		
HTTPS	TCP	443	모든 IPv4 주소		
사용자 지정	TCP	8000	모든 IPv4 주소		

- 이제 <https://<도메인네임>> 으로 접속확인을 해 본다.

8. Forever 설치

- node server.js 로 서버를 구동했을 경우, 터미널이 닫히면, 서버 구동이 중지된다.
- 따라서, 지속적으로 백그라운드로 Node Server를 구동하기 위해서는 Forever가 필요하다.

◆ Forever 설치 및 구동

sudo npm install forever -g	# 설치
sudo forever start server.js	# 구동
sudo forever list	# 구동중인 프로그램 list
sudo forever stop 0	# 0번 프로세스 중지

- 이제 forever를 구동하고 터미널을 종료해도 계속 실행된다.