

Nginx를 Reverse Proxy로 사용하여 HTTPS 설정하고 Spring Web 애플리케이션 연결하기

- https 설정은 도메인 네임이 설정된 사이트에 설정할 수 있으므로 사이트에 도메인 네임을 설정한다
- 한개의 EC2 인스턴스에 Nginx와 docker기반에서 Spring-web 컨테이너를 실행하는 구조
- Nginx는 80포트, Spring-web은 8081포트 사용
- Nginx는 도커를 사용하지 않고 EC2 인스턴스에 설치하고 실행(80포트 사용)
- 이용자가 http 접속하면 https로 전환함
- 이용자는 <https://domain.name:80> 으로 요청하면 <http://domain.name:8081> 으로 다시 요청한다
- 예) <https://domain.name/add/3/5> --> <http://domain.name:8081/add/3/5>

EC2 인스턴스에 Nginx 설치

- `sudo apt update`
- `sudo apt install -y nginx`
- `sudo systemctl start nginx`
- `sudo systemctl enable nginx`
- 웹브라우저에서 테스트, <http://domain.name>
- Nginx의 초기 화면이 표시되면 된다

Certbot 설치

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

HTTPS 인증서 발급

- `sudo certbot certonly --webroot -w /var/www/html -d your.domain -d www.your.domain`
- 발급된 인증서 확인 발급된 인증서는 `/etc/letsencrypt/live/ai-item.shop/`에 저장됨

Nginx 설정 업데이트

- `sudo nano /etc/nginx/sites-available/default`
- 아래의 내용으로 변경한다

HTTP 요청 처리 (80 포트 -> HTTPS 리디렉션)

```
server {  
    listen 80 default_server;  
    listen [::]:80 default_server;  
  
    server_name ai-item.shop www.ai-item.shop;  
  
    # Let's Encrypt 인증을 위한 설정  
    location /.well-known/acme-challenge/ {  
        root /var/www/html;  
    }  
  
    # HTTP -> HTTPS 리디렉션  
    return 301 https://$host$request_uri;  
}
```

HTTPS 요청 처리 (443 포트)



```

server {
    listen 443 ssl;
    listen [::]:443 ssl ipv6only=on;

    server_name ai-item.shop www.ai-item.shop;

    # SSL 인증서 설정
    ssl_certificate /etc/letsencrypt/live/ai-item.shop/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/ai-item.shop/privkey.pem;
    include /etc/letsencrypt/options-ssl-nginx.conf;
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem;

    # Spring 애플리케이션 리버스 프록시 설정
    location / {
        proxy_pass http://localhost:8081;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }
}

```

- sudo nginx -t
- sudo systemctl reload nginx

인증서 자동 갱신 설정

- sudo certbot renew --dry-run
- sudo systemctl list-timers | grep certbot

HTTPS 설정 확인

- <http://domain.name/> 으로 요청했을 때 https 으로 변경되면서 spring-web 으로 연결되는지 확인
- 이후에는 모든 https 요청에 spring-web 이 실행되는지 확인