

---

# Server Deployment Log

**Domain:** justsaying.co.kr

**Date:** 2025-05-05

---

## 1. Nginx 설정 현황

현재 Nginx는 HTTP 요청을 HTTPS로 리디렉션하고, SSL 인증서를 통해 보안을 유지하고 있음.

또한 다음과 같은 경로별 프록시 설정이 적용되어 있음.

### HTTP → HTTPS 리디렉션

```
server {  
  
    listen 80;  
  
    server_name justsaying.co.kr;  
  
    return 301 https://$host$request_uri;  
  
}
```

•

### HTTPS 설정 및 프록시 경로

```
server {  
  
    listen 443 ssl;  
  
    server_name justsaying.co.kr;  
  
  
    ssl_certificate /etc/letsencrypt/live/justsaying.co.kr/fullchain.pem;  
  
    ssl_certificate_key /etc/letsencrypt/live/justsaying.co.kr/privkey.pem;  
  
    include /etc/letsencrypt/options-ssl-nginx.conf;
```

```
ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem;
```

```
location /api/ {  
    add_header 'Access-Control-Allow-Origin' '*' always;  
    proxy_pass http://localhost:8080/  
}
```

```
location /api-translator/ {  
    proxy_pass http://localhost:8001;  
}
```

```
location / {  
    proxy_pass https://d285wb9kugqt55.cloudfront.net;  
    proxy_ssl_verify on;  
}  
}
```

•

---

## 2. Spring Boot 서버 상태

- EC2 인스턴스에서 `demo_backend-0.0.1-SNAPSHOT.jar` 파일이 정상 실행 중.
- 포트 8080에서 작동하며, Kakao 로그인 경로(`/api/kakao/login`)는 JWT 인증 필터에서 예외 처리됨.
- 모든 API 응답이 정상적으로 반환됨

---

## 3. 문제 발생 정리

### 3.1. Kakao 로그인 에러

- 에러 메시지:  
`Uncaught TypeError: event.data.substring is not a function`
- API 요청:  
`POST /api/kakao/login`
- 응답 상태:  
`200 OK, Content-Type: text/html`
- Server: AmazonS3
- 문제 요약  
JSON 응답이 와야 하는데 HTML 응답이 옴 → CloudFront에서 캐시된 HTML 파일이 반환된 것으로 추정됨

### 3.2. 로컬에서는 정상 동작

- 로컬 개발 환경에서는 정상적으로 JSON 응답이 반환됨.
- 배포환경(CloudFront + Nginx)에서는 HTML이 반환됨, 프론트에서 JSON으로 파싱하지 못해 에러 발생.

---

## 4. 추정 원인 및 조치 필요 항목

- CloudFront가 `/api/*` 요청도 정적 파일로 잘못 인식해 HTML을 반환하는 것으로 추정됨.
- Nginx의 `proxy_set_header` 누락 가능성 있음.
- `.env.production` 내 API 경로 설정 확인 필요.
- Spring Boot 로그에 추가 정보를 남겨야 원인 분석이 쉬움.

---

## 5. 서버 로그 상태

### 5.1. access.log

- `/api/kakao/login` 요청 기록 다수 존재.
- 응답 코드가 403, 400, 301 등 혼재됨.

## 5.2. error.log

- 현재까지 특별한 에러 로그는 없음.

## 5.3. Spring Boot 로그

- `UserMapper`, `DiaryMapper` 등 정상 주입 확인.
  - JWT 인증 필터 작동 중이며, 로그인 및 세션 처리 완료됨.
- 

## 6. 기타 참고 사항

- EC2에서 Java 프로세스 2개 실행 중.
    - 하나는 포트 8080에서 작동.
    - 다른 하나는 포트 미지정 상태로 중복 실행 가능성 있음.
    - 중복 실행 여부 점검 필요.
- 

## 7. 다음 작업 제안

- CloudFront의 `/api/*` 경로 캐시 제외 처리.
- - Nginx 설정 점검 및 리다이렉션/헤더 수정
  - Nginx의 `proxy_set_header` 설정 점검.
- `.env.production`의 `VITE_APP_API_BASE_URL` 값 확인.
- `/api/kakao/login` 요청에 대한 Spring Boot 로그 추가

---

## [자동 실행 스크립트: **start.sh**]

```
#!/bin/bash
```

```
# 실행할 JAR 파일 이름
```

```
JAR_NAME="demo_backend-0.0.1-SNAPSHOT.jar"
```

```
# 실행 중인 프로세스 확인
```

```
PID=$(ps -ef | grep $JAR_NAME | grep -v grep | awk '{print $2}')
```

```
if [ -z "$PID" ]; then
```

```
    echo "기존 실행 없음. 서버 시작 중..."
```

```
    nohup java -jar $JAR_NAME > logs/server.log 2>&1 &
```

```
    echo "$JAR_NAME 실행 완료"
```

```
else
```

```
    echo "이미 실행 중인 프로세스가 있음. 중복 실행 방지됨 (PID: $PID)"
```

```
fi
```

---

## [사용법]

### 1. 실행 권한:

```
chmod +x start.sh
```

### 2. 실행 :

```
./start.sh
```

---

## [로그 저장 위치]

- **logs/server.log** 파일에 서버 로그가 저장돼.
- **nohup**을 사용했기 때문에 **EC2** 세션 종료 후에도 서버가 꺼지지 않아.

---

## [서버 종료 스크립트: **stop.sh**]

```
#!/bin/bash
```

```
# 종료할 JAR 파일 이름
```

```
JAR_NAME="demo_backend-0.0.1-SNAPSHOT.jar"
```

```
# 실행 중인 프로세스 ID 검색
```

```
PID=$(ps -ef | grep $JAR_NAME | grep -v grep | awk '{print $2}')
```

```
if [ -z "$PID" ]; then
```

```
    echo "실행 중인 프로세스가 없습니다. 종료할 서버가 없습니다."
```

```
else
```

```
    echo "서버 종료 중... (PID: $PID)"
```

```
    kill -15 $PID
```

```
    echo "서버 종료 완료"
```

```
fi
```

---

## [사용법]

1. EC2에서 **stop.sh**로 저장해.

실행 권한을 부여 : `chmod +x stop.sh`

2. 서버 종료:

```
./stop.sh
```

---

## [추가 설명]

- `kill -15`는 안전한 종료(**SIGTERM**) 명령 (가능하면 `-9` 같은 강제 종료보다 더 낫다.)
- 서버가 로그 파일을 기록 중일 때, 기회를 주고 종료시킬 수 있다.
- JAR 이름은 `start.sh`와 반드시 동일하게 맞춰야 된다.