Nginx + TLS 설정

설치파일 & 방화벽(UFW) 열기

```
sudo apt-get update
sudo apt-get install -y nginx certbot python3-certbot-nginx

sudo ufw allow 80/tcp
sudo ufw allow 443/tcp
```

Nginx 기본 설정 (HTTP만 먼저)

인증서 발급 전에는 80 포트만 열어두는 게 안전해.

프론트 정적파일을 직접 서빙하고, `/api/v1`는 FastAPI로 프록시.

```
sudo nano /etc/nginx/conf.d/visioninapp.conf
```

```
map $http_upgrade $connection_upgrade { default upgrade; '' close; }

server {
  listen 80;
  server_name k13s303.p.ssafy.io;

  # 인증서 발급용(LE) 챌린지 파일 경로
  location /.well-known/acme-challenge/ { root /var/www/certbot; }

  # 프론트 정적 파일(바닐라 JS)
  root /home/ubuntu/apps/frontend/current;
  index index.html;

  # SPA 라우팅 고려
  location / {
    try_files $uri $uri/ /index.html;
  }
```

```
  # FastAPI 프록시 (uvicorn: 127.0.0.1:8000, systemd로 관리 중)
  location /api/v1/ {
    proxy_http_version 1.1;
    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;

    proxy_pass http://127.0.0.1:8000;
  }

  # 캐싱/압축 — 정적 자원 응답 최적화
  location ~* \.(?:js|css|png|jpg|jpeg|gif|svg|ico|woff2?)$ {
    expires 7d;
    add_header Cache-Control "public";
  }
}
```

적용:

```
sudo nginx -t && sudo systemctl reload nginx
```

브라우저로 `http://your.domain.or.ip` 열어서 프론트가 나오는지 먼저 확인!

---

3) TLS 발급(Certbot, nginx 플러그인)

```
sudo certbot --nginx -d your.domain.or.ip --redirect
```

- 완료되면 **자동으로 443 서버블록이 추가**되고, 80은 HTTPS로 **리다이렉트**됨.
- 만약 certbot이 블록을 새로 만들어주지 않았다면 아래 443 블록을 수동으로 추가하면 돼.

---

4) 최종 TLS 서버블록(수동 구성 시)

위 2번의 80 블록은 **리다이렉트 전용**으로 바꾸고, 443에 실제 서비스를 둔다.

```
sudo nano /etc/nginx/conf.d/visioninapp.conf
```

```
map $http_upgrade $connection_upgrade { default upgrade; '' close; }

# ──────────────────────────────────────────────────────────────────────
# 80: 모두 HTTPS로 리다이렉트 + ACME 챌린지
# ──────────────────────────────────────────────────────────────────────
server {
  listen 80;
  listen [::]:80;
  server_name {EC2_Domain};

  # certbot HTTP-01 챌린지
  location /.well-known/acme-challenge/ { root /var/www/certbot; }

  # 나머지는 전부 HTTPS로
  return 301 https://$host$request_uri;
}

# ──────────────────────────────────────────────────────────────────────
# 443: 실제 서비스 (SPA + /api/v1 프록시)
# ──────────────────────────────────────────────────────────────────────
server {
  listen 443 ssl http2;
  listen [::]:443 ssl http2;
  server_name k13s303.p.ssafy.io;

  # SSL (Certbot 경로)
  ssl_certificate     /etc/letsencrypt/live/{EC2_Domain}/fullchain.pem;
  ssl_certificate_key /etc/letsencrypt/live/{EC2_Domain}/privkey.pem;
  include /etc/letsencrypt/options-ssl-nginx.conf;
  ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem;
```

```
  # 공통 프록시 헤더
  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;

  # 프론트 정적 (바닐라 JS / SPA)
  root /home/ubuntu/apps/frontend/current;
  index index.html;

  location / {
    try_files $uri $uri/ /index.html;
  }

  # FastAPI: /api/v1 → 127.0.0.1:8000
  location /api/v1/ {
    proxy_http_version 1.1;
    proxy_read_timeout 60s;

    proxy_pass http://127.0.0.1:8000;
  }

  # 정적 리소스 캐시
  location ~* \.(?:js|css|png|jpg|jpeg|gif|svg|ico|woff2?)$ {
    expires 7d;
    add_header Cache-Control "public";
  }

  access_log /var/log/nginx/visioninapp.access.log;
  error_log  /var/log/nginx/visioninapp.error.log;
}
```

적용:

```
# nginx가 js 파일을 읽을 수 있도록 실행 권한 부여
sudo setfacl -m u:www-data:x /home /home/ubuntu /home/ubuntu/apps /h
ome/ubuntu/apps/frontend /home/ubuntu/apps/frontend/current
```

```
sudo nginx -t && sudo systemctl reload nginx
```

---

5) 인증서 갱신 리허설

```
sudo certbot renew --dry-run
```