Edu Network PoC #1보고서

**프로젝트명:** Edu Network PoC #1

**보고 일자:** 2025-10-12

**상태:** 성공

# 1. 프로젝트 아키텍처

PoC 환경은 다음과 같은 주요 구성 요소로 이루어져 있습니다.

* Origin Server (edu-api**):** Python FastAPI 기반의 원본 서버로, 원격(라즈베리파이)에서 실행되며 테스트용 비디오 파일(dummy.mp4)을 제공합니다.
* Client (edu-web-tablet**):** React 기반의 웹 애플리케이션으로, 사용자에게 파일 다운로드 UI를 제공하고 다운로드 속도, 시간 등 실시간 통계를 표시합니다.
* Forward & Caching Proxy (nginx**):** 클라이언트(macOS) 측에서 Docker를 통해 edu-web-tablet과 함께 실행되며, 포워드 프록시 및 캐싱 계층 역할을 수행합니다.

# 2. 검증 환경 및 방법

## 2.1. 환경 구성

본 기술 검증은 실제 하드웨어 환경에서 다음과 같이 구성하여 수행되었습니다.

* **서버:** 라즈베리파이(Raspberry Pi)에서 Origin Server(edu-api)를 원격으로 호스팅했습니다.
* **클라이언트:** macOS 환경에서 Docker를 사용하여 웹 애플리케이션(edu-web-tablet)과 포워드 프록시(nginx)를 함께 실행하고, 운영체제 네트워크 설정에 PAC(Proxy Auto-Config) 를 적용하여 프록시를 자동 설정한 후 접속하여 테스트를 진행했습니다.

## 2.2. 검증 시나리오

검증은 다음 시나리오에 따라 진행되었습니다.

**1.** 클라이언트가 웹 UI에서 "Start Download" 버튼을 클릭하여 파일 다운로드를 5회 연속으로 요청합니다.

**2.** 최초 요청 시, Nginx 프록시는 Origin 서버(edu-api)로부터 파일을 받아와 클라이언트에 전송하고, 해당 파일을 캐시에 저장합니다.

**3.** 두 번째 요청부터는 Nginx 프록시가 Origin 서버에 재요청하지 않고, 캐시에 저장된 파일을 직접 클라이언트에 전송합니다.

**4.** 웹 UI에 표시되는 다운로드 시간과 속도를 통해 캐싱 효과를 정량적으로 확인합니다.

# 3. 검증 결과

## 3.1. 기능 및 성능 검증

실제 검증 결과, 캐싱 프록시의 기능이 정상적으로 동작함을 확인했습니다. 최초 다운로드 대비 **이후의 모든 다운로드 속도가 현저하게 향상**되었으며, 다운로드 시간 또한 크게 단축되었습니다. 이를 통해 캐싱 프록시가 반복적인 콘텐츠 요청에 대해 네트워크 효율성을 극대화할 수 있음을 성공적으로 입증했습니다.

## 3.2. 부하 테스트 (k6)

서버의 안정성을 확인하기 위해 k6를 사용하여 부하 테스트를 수행했습니다. 테스트 조건은 다음과 같습니다.

export const options = {  
 vus: 100, // 100명의 가상 사용자  
 iterations: 500, // 총 500회 요청 (사용자당 5회)  
  
 duration: '10m', // 최대 10분 내 완료  
};

테스트 결과, 100명의 가상 사용자가 동시에 요청하는 부하 상황에서도 클라이언트(macOS)에서 실행된 **Nginx 프록시 서버의 최대 CPU 사용량은 약 9% 수준**(단일 코어 100% 기준)으로 매우 낮게 유지되었습니다. 이는 Nginx 캐싱 프록시가 적은 리소스로도 높은 처리량을 감당할 수 있음을 의미하며, 솔루션의 안정성과 효율성을 뒷받침합니다.

# 4. 결론

본 PoC를 통해 Nginx를 이용한 캐싱 프록시 도입이 네트워크 성능 개선에 매우 효과적임을 성공적으로 검증했습니다. 특히, 클라이언트 환경에서 실행된 Nginx가 부하 테스트 시 낮은 CPU 점유율을 기록하며 안정적인 서비스를 제공할 수 있음을 확인했습니다.

따라서 동일한 콘텐츠를 다수의 사용자에게 반복적으로 제공해야 하는 교육용 네트워크 환경에서 캐싱 프록시는 **서버 부하 감소, 네트워크 트래픽 절감, 사용자 경험 향상**을 위한 효율적이고 실용적인 솔루션이 될 수 있습니다.