

# Compression

📅 발표 날짜	@2021년 9월 11일
☑ 발표 여부	☑
≡ 발표자	이연중

압축은 응답 본문의 크기를 줄여주어 웹 앱의 속도를 높일 수 있다.

프로덕션 환경에서 트래픽이 많은 웹의 경우 어플리케이션 서버(일반적으로 역 프록시)에서 압축을 오프로드하는 것이 좋다.

이 경우, 압축 미들웨어를 사용하면 안된다.

## 역 프록시(reverse proxy)?

forward proxy(default)가 서버 -> 프록시 서버 -> 브라우저라면,

reverse proxy는 브라우저 -> 프록시 서버 -> 서버

브라우저로부터 받은 요청을 프록시 서버가 수신하고, 서버에서 받은 응답을 브라우저로 전달하는 중계자.

서버는 이러한 프록시 서버에 압축을 오프로딩하는 것이다.(압축을 프록시 서버한테 맡긴다)

## Express에서의 사용(Default)

압축 미들웨어 패키지를 사용해 gzip 압축을 활성화한다.

```
npm install --save compression
```

그 다음, 압축 미들웨어를 글로벌 미들웨어로 적용한다.

```
//main.ts
import * as compression from 'compression';

app.use(compression());
```

## Fastify에서의 사용

`FastifyAdapter`를 사용하는 경우 fastify-compress를 사용하는 것이 좋다.

```
npm install --save fastify-compress
```

그 다음, fastify-compress 미들웨어를 글로벌 미들웨어로 적용한다.

```
import compression from 'fastify-compress';  
  
app.register(compression);
```

node -v >= 11.7.0에서 fastify-compress는 기본적으로 Brotli 압축을 사용한다.

Brotli는 압축률은 좋지만 속도가 느리다.

fastify-compress에 gzip 및 deflate 인코딩만 사용하도록 해 응답속도를 빨리할 수 있다.  
(Brotli보다 압축률은 떨어짐)

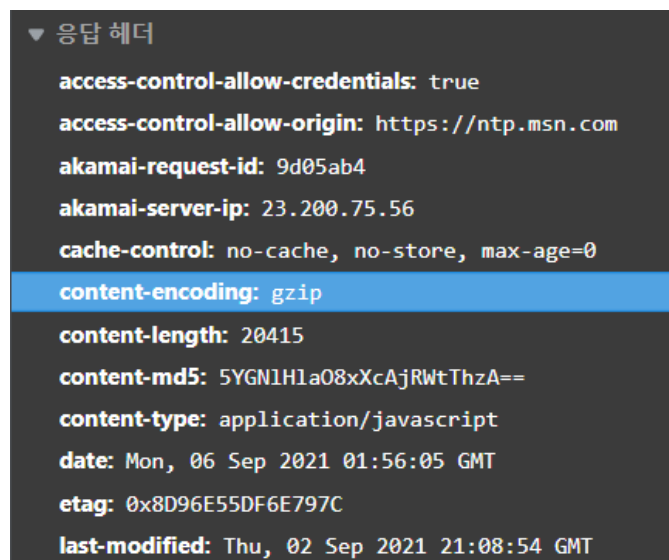
```
app.register(compression, { encodings: ['gzip', 'deflate'] });
```

gzip과 deflate 모두를 클라이언트가 지원하는 경우에는 gzip을 우선시한다.

## 어떤 인코딩을 지원하는지 확인하는 방법

Dev Tools -> Network Tab -> Response Header Check!

### MS Edge



### Google Chrome

#### ▼ Response Headers

**accept-ranges:** bytes

**age:** 57882

**alt-svc:** h3=":443"; ma=2592000,h3-29=":443"; ma=2592000,h3-T053"; ma=2592000; v="46,43"

**cache-control:** public, immutable, max-age=31536000

**content-encoding:** gzip

**content-length:** 58773

**content-security-policy-report-only:** require-trusted-types-for 's

**content-type:** text/javascript; charset=UTF-8

**cross-origin-resource-policy:** cross-origin

**date:** Tue, 31 Aug 2021 09:01:31 GMT

**expires:** Wed, 31 Aug 2022 09:01:31 GMT

**last-modified:** Mon, 30 Aug 2021 02:53:55 GMT