Apache□ □□ □□□□

- [SECURITY] http header□ cookie
- [Security] Content-Security-Policy
- [SECURITY] Header □□□□□

[SECURITY] http header의 cookie

쿠키 이름	역할	**Secure** 플래그	**HttpOnly** 플래그
`PHPSESSID`	PHP 기본 세션 쿠키	적용 안됨	적용 안됨
`XSRF-TOKEN`	CSRF 보호를 위한 토큰	적용됨	적용 안됨
`Laravel 세션 쿠키:`	Laravel 세션 쿠키	적용됨	적용됨

XSRF-ROKEN과 Laravel 세션 쿠키는 프론트에서 처리 가능

> ## **PHPSESSID**의 **Secure** 플래그 추가 방법

- ### `php.ini` 설정 수정

  ```
  session.cookie_secure = On
  ```

- ### **PHP** 코드에서 설정

  `session_start()`를 호출하기 전에 아래와 같이 설정합니다:

  ```
  ini_set('session.cookie_secure', 1);
  ini_set('session.cookie_httponly', 1); // HttpOnly 추가
  session_start();
  ```

- ### `.htaccess` 파일에서 설정

- `.htaccess`를 사용하여 PHP 설정을 수정할 수 있습니다:

```
php_value session.cookie_secure On
php_value session.cookie_httponly On
```

[Security] Content-Security-Policy

Content-Security-Policy header is not set. It restricts resources the page can load and prevents XSS attacks.

seo 진단시 위와 같은 진단내용이 잡힌다면 아래 내용들을 확인해주세요.

> ## Content-Security-Policy(CSP)란?

CSP는 HTTP 응답 헤더 중 하나로, 브라우저가 페이지에 어떤 리소스를 로드할 수 있는지 제어하는 보안 정책을 정의합니다.
이를 통해 아래와 같은 보안 위협을 방지할 수 있습니다:

1. **XSS(크로스 사이트 스크립팅)**
   공격자가 악성 JavaScript를 주입하는 것을 방지합니다.
2. 데이터 주입 공격
   악성 리소스(URL, CSS, 폰트 등)의 로드를 제한합니다.
3. **Clickjacking**
   악성 iframe 삽입 등을 방지합니다.

> ## Content-Security-Policy가 중요한 이유

CSP가 없으면 다음과 같은 문제가 발생할 수 있습니다:

- 외부에서 [illegible]를 주입하여 사
- 비인가된 리소스가 로드되어 정보 유출 가능.
- HTTPS 환경에서도 보안이 취약한 HTTP 리소스가 로드될 가능성.

□□ □□: Content-Security-Policy □□ □□

Apache □□□ □□ CSP □□

Apache[`Content-Security-Policy` □[`.htaccess` □□□ `httpd.conf` , `apache2.conf`)□ □□ □□□ □□□□□.

```
<IfModule mod_headers.c>
Header set Content-Security-Policy "default-src 'self'; script-src 'self' https://www.googletagmanager.com
https://d3js.org https://cdnjs.cloudflare.com https://cdn.jsdelivr.net 'unsafe-inline' 'unsafe-eval'; style-src 'self'
https://fonts.googleapis.com https://cdn.jsdelivr.net 'unsafe-inline'; font-src 'self' https://fonts.gstatic.com
https://cdn.jsdelivr.net data:; connect-src 'self' https://www.google-analytics.com; img-src 'self' data:
https://cdn.jsdelivr.net;"
</IfModule>
```

> default-src 'self'

□□ □□□ □□ □□ □□□□ □□□(□□□□, □□□, □□□)□ □□□□ □□□ □□□□□.

> self

□□ □□□ □□□□ □□□□ □□□□□ □□□□□.

> unsafe-inline

□□□ □□□□□ □□□□□ (□□□ □□□□ □□).

> unsafe-eval

`eval()` [`setTimeout(string)` □□ □□ □□ □□□ □□□□□.

> data:

□□□ URI□ □□ □□, □□□ □□ □□□□ □□□□□.

* □□□□ □□□□ 500 server error□ □□□ □ □□□□.

2. □□□□□ □□□□ □□ (□□□ □□)

1. app/Http/Middleware□ □□□ class□ □□□□□.

2. □□□ □□ handle function□ □□□□□. $csp □□ □□ □□□ □□ □□□□□□□□□.

```
public function handle(Request $request, Closure $next)
    {
        $response = $next($request);



        $csp = "default-src 'self'; " .
       "script-src 'self' https://www.googletagmanager.com https://d3js.org https://cdnjs.cloudflare.com
https://cdn.jsdelivr.net 'unsafe-inline' 'unsafe-eval'; " .
       "style-src 'self' https://fonts.googleapis.com https://cdn.jsdelivr.net 'unsafe-inline'; " .
       "font-src 'self' https://fonts.gstatic.com https://cdn.jsdelivr.net data:; " .
       "connect-src 'self' https://www.google-analytics.com; " .
       "img-src 'self' data: https://cdn.jsdelivr.net;";



        $response->headers->set('Content-Security-Policy', $csp);



        return $response;
    }
```

3. `app/Http/Kernel.php` □ □□□□□ □□□□□.

```
protected $middleware = [ \App\Http\Middleware\ContentSecurityPolicy::class, ];
```

□□□

□□□ □□□□ □□

- [unsafe-inline]□ □□□ □ □□□, □□ □□□ □□□□ □□ □□□□.
- nonce [hash] □□ CSP□ □□□□ □□ □□□□□ □□□□□ □□□□ □□ □□□□.

Nonce□ □□ (□□)

□□ □□□ nonce □□□ □□ □□□□□□□.

- apache □□

```
Header set Content-Security-Policy "default-src 'self'; script-src 'self' https://www.googletagmanager.com
https://d3js.org https://cdnjs.cloudflare.com 'nonce-abc123'; style-src 'self' https://fonts.googleapis.com
https://cdn.jsdelivr.net 'unsafe-inline'; font-src 'self' https://fonts.gstatic.com; connect-src 'self'
https://api.example.com; img-src 'self' data:;"
```

- script □□□□

```
<script nonce="abc123"> console.log('Inline script with nonce'); </script>
```

- □□ Nonce □□

```
<?php
// php + javascript
$nonce = base64_encode(random_bytes(16));
header("Content-Security-Policy: script-src 'self' 'nonce-$nonce'");
?>

<script nonce="<?php echo $nonce; ?>">
  console.log('This script uses a nonce');
</script>
```

```
<script>
  // only javascript
  // Generate a random nonce
  const array = new Uint8Array(16);
```

```
  window.crypto.getRandomValues(array);
  const nonce = btoa(String.fromCharCode.apply(null, array));

  // Set the Content-Security-Policy header
  const meta = document.createElement('meta');
  meta.setAttribute('http-equiv', 'Content-Security-Policy');
  meta.setAttribute('content', `script-src 'self' 'nonce-${nonce}'`);
  document.head.appendChild(meta);

  // Dynamically create a script element with the nonce
  const script = document.createElement('script');
  script.setAttribute('nonce', nonce);
  script.textContent = "console.log('This script uses a nonce')";
  document.body.appendChild(script);
</script>
```

[SECURITY] Header □□□□□

Header	OK	Notice	Warning	Critical	Recommendation
X-Frame-Options	0	0	77	0	X-Frame-Options header is not set. It prevents clickjacking attacks when set to 'deny' or 'sameorigin.'
X-Content-Type-Options	0	0	77	0	X-Content-Type-Options header is not set. It stops MIME type sniffing and mitigates content type attacks.
Referrer-Policy	0	0	77	0	Referrer-Policy header is not set. It controls referrer header sharing and enhances privacy and security.
Feature-Policy	0	0	77	0	Feature-Policy header is not set. It allows enabling/disabling browser APIs and features for security. Not important if Permissions-Policy is set.
Permissions-Policy	0	0	77	0	Permissions-Policy header is not set. It allows enabling/disabling browser APIs and features for security.
Server	0	0	77	0	Server header is set to known 'Apache.' It is better not to reveal used technologies.
Set-Cookie	70	65	70	0	Set-Cookie header for 'PHPSESSID' does not have 'SameSite' flag. Consider using 'SameSite=Strict' or 'SameSite=Lax.'
					Set-Cookie header for 'XSRF-TOKEN' does not have 'HttpOnly' flag. Attacker can steal the cookie using XSS. Consider using 'HttpOnly' when cookie is not used by JavaScript.

```
--------------------------------------------------------------------------------
```

> “
>
> # X-Frame-Options

□□:

- X-Frame-Options □□□ □□□□ □□.

□□:

Apache □□ □□□ □□□ □□:

```
Header always set X-Frame-Options "DENY"
```

- **DENY**: iframe □□□ □□□ □□.
- **SAMEORIGIN**: □□ □□□□□□ iframe □□ □□.

> “
>
> # X-Content-Type-Options

□□:

- X-Content-Type-Options □□□ □□□□ □□.

□□:

Apache □□ □□□ □□□ □□:

```
Header always set X-Content-Type-Options "nosniff"
```

- MIME □□ □□□ □□.

>

Referrer-Policy

□□:

- Referrer-Policy □□□ □□□□ □□.

□□:

Apache □□ □□□ □□□ □□:

```
Header always set Referrer-Policy "no-referrer"
```

- **no-referrer**: □□ □□□ □□□ □□.
- □□ `strict-origin`, `strict-origin-when-cross-origin`, `same-origin` □.

> “
Feature-Policy / Permissions-Policy

□□:

- Feature-Policy □□ Permissions-Policy □□□ □□□□ □□.

□□:

Apache □□ □□□ □□□ □□:

```
Header always set Permissions-Policy "geolocation=(), camera=(), microphone=()"
```

- **Permissions-Policy**: □□ □□□□ □□(API) □□□ □□.
- □: `geolocation`, `camera`, `microphone` □.

>

Server

문제:

- Server 헤더가 Apache로 설정되어 서버 정보가 노출됨.

해결:

Apache 설정 파일에 다음을 추가:

```
ServerTokens Prod
ServerSignature Off
Header unset Server
```

- **ServerTokens Prod**: 최소한의 정보만 노출.
- **ServerSignature Off**: 오류 페이지에서 Apache 정보 숨김.
- **Header unset Server**: Server 헤더를 제거.

Set-Cookie

문제:

- `SameSite` [`HttpOnly` 플래그가 없음.

해결:

Apache 설정 파일에 다음을 추가:

```
Header always edit Set-Cookie ^(.*)$ "$1; SameSite=Strict; HttpOnly; Secure"
```

- **SameSite=Strict**: 쿠키를 외부 요청에 전송하지 않음.
- **HttpOnly**: JavaScript에서 쿠키 접근을 차단.
- **Secure**: HTTPS 요청에서만 쿠키를 전송.

□□ □□□

```
# X-Frame-Options □□
Header always set X-Frame-Options "DENY"

# X-Content-Type-Options □□
Header always set X-Content-Type-Options "nosniff"

# Referrer-Policy □□
Header always set Referrer-Policy "no-referrer"

# Permissions-Policy □□
Header always set Permissions-Policy "geolocation=(), camera=(), microphone=()"

# Server □□ □□□
ServerTokens Prod
ServerSignature Off
Header unset Server

# Set-Cookie □□ □□
Header always edit Set-Cookie ^(.*)$ "$1; SameSite=Strict; HttpOnly; Secure"
```