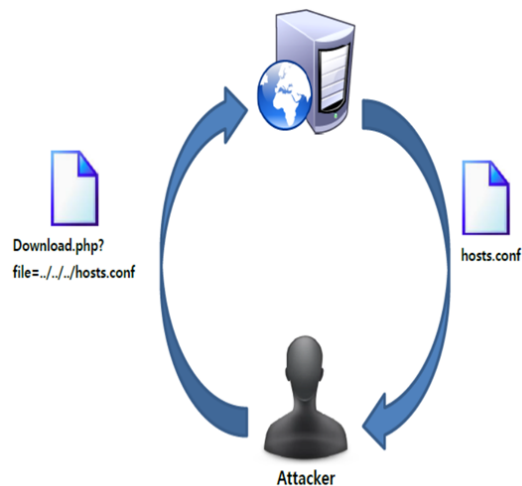


# File-download-1

| web

## File Download 취약점

- 웹 서버 내의 파일을 다운로드 할 때, **경로를 임의로 조작하여** 클라이언트 영역에서 볼 수 없는 파일들을 다운로드 시도하는 취약점



### 일반적인 사용자

- 경우 웹 서버에게 정상적인 파일의 경로를 전송하여 요청

### 공격자

- 정상적인 파일의 경로를 입력하는 것이 아닌 공격자가 원하는 파일의 경로를 웹 서버에게 전달

## 문제

### 문제 설명

File Download 취약점이 존재하는 웹 서비스입니다.  
flag.py를 다운로드 받으면 플래그를 획득할 수 있습니다.

Reference

[Introduction of Webhacking](#)

[Translate](#)

## 풀이

Please upload your memo!

서버를 열면 위와 같은 화면이 나온다.

## Upload Your Own Memo

Filename

Content

Upload

**[Upload My Memo]** 를 누르면 위와 같은 화면이 나온다.

## Upload Your Own Memo

Filename

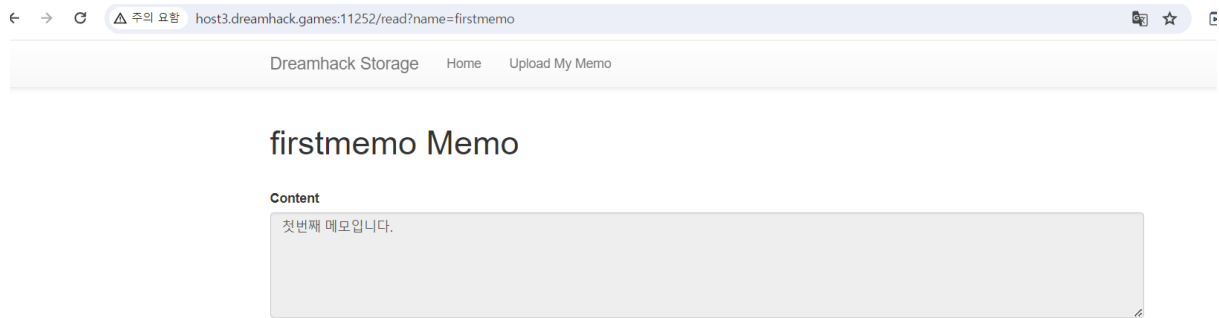
Content

Upload

메모를 업로드 해보자.

## Your uploaded memos

- [firstmemo](#)



**firstmemo** 메모가 작성된 것을 확인하였다.

이 때, 페이지의 경로가 **/read?name=firstmemo**임을 볼 수 있다.

#### <문제 코드>

```
#!/usr/bin/env python3
import os
import shutil

from flask import Flask, request, render_template, redirect

from flag import FLAG

APP = Flask(__name__)

UPLOAD_DIR = 'uploads'

@APP.route('/')
def index():
    files = os.listdir(UPLOAD_DIR)
    return render_template('index.html', files=files)

@APP.route('/upload', methods=['GET', 'POST'])
def upload_memo():
    if request.method == 'POST':
        filename = request.form.get('filename')
        content = request.form.get('content').encode('utf-8')

        if filename.find('.') != -1:
            return render_template('upload_result.html', data='bad characters,,')

        with open(f'{UPLOAD_DIR}/{filename}', 'wb') as f:
            f.write(content)

        return redirect('/')

    return render_template('upload.html')
```

```

@APP.route('/read')
def read_memo():
    error = False
    data = b''

    filename = request.args.get('name', '')

    try:
        with open(f'{UPLOAD_DIR}/{filename}', 'rb') as f:
            data = f.read()
    except (IsADirectoryError, FileNotFoundError):
        error = True

    return render_template('read.html',
                           filename=filename,
                           content=data.decode('utf-8'),
                           error=error)

if __name__ == '__main__':
    if os.path.exists(UPLOAD_DIR):
        shutil.rmtree(UPLOAD_DIR)

    os.mkdir(UPLOAD_DIR)

    APP.run(host='0.0.0.0', port=8000)

```

#### <upload\_memo 함수>

```

@APP.route('/upload', methods=['GET', 'POST'])
def upload_memo():
    if request.method == 'POST':
        filename = request.form.get('filename')
        content = request.form.get('content').encode('utf-8')

        if filename.find('.') != -1:
            return render_template('upload_result.html', data='bad characters,,')

        with open(f'{UPLOAD_DIR}/{filename}', 'wb') as f:
            f.write(content)

        return redirect('/')

    return render_template('upload.html')

```

**upload\_memo** 함수를 보면 **filename**을 입력할 때 **..**와 함께 입력될 경우 **'bad characters,,'** 라는 문구를 나타냄을 알 수 있다.

Dreamhack Storage   Home   Upload My Memo

## Upload Your Own Memo

**Filename**

**Content**  

flag.py

Upload

Dreamhack Storage   Home   Upload My Memo

## Raw Socket Sender Result

bad characters,,

[Back](#)

한 번 시도해보았고 예상대로 **bad characters,,** 라는 문구가 나오는 것을 확인했다.

### <read 함수>

```
@APP.route('/read')
def read_memo():
    error = False
    data = b''

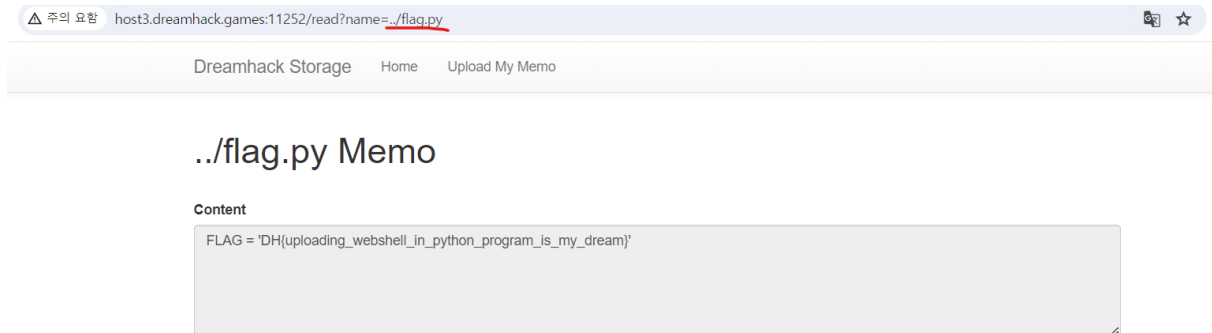
    filename = request.args.get('name', '')

    try:
        with open(f'{UPLOAD_DIR}/{filename}', 'rb') as f:
            data = f.read()
    except (IsADirectoryError, FileNotFoundError):
        error = True

    return render_template('read.html',
                           filename=filename,
                           content=data.decode('utf-8'),
                           error=error)
```

**read** 함수에서는 **upload\_memo** 함수와는 달리 **filename**에 대한 필터링 없이 파일 내용을 읽을 수 있다.  
**{UPLOAD\_DIR}/{filename}**을 통해 업로드 된 파일들의 위치가 **filename**의 바로 상위 디렉터리임을 알 수 있다.

따라서 아까 URL에서 본 `/read?name=firstmemo` 부분을 `/read?name=../flag.py`로 바꾸어주면 **flag.py** 파일에 접근할 수 있을 것이다.



URL을 변경해 주었더니 FLAG 값을 획득할 수 있었다.

풀이 228



대단해요! 정답을 맞추셨네요. 문제를 어떻게 해결하셨나요?

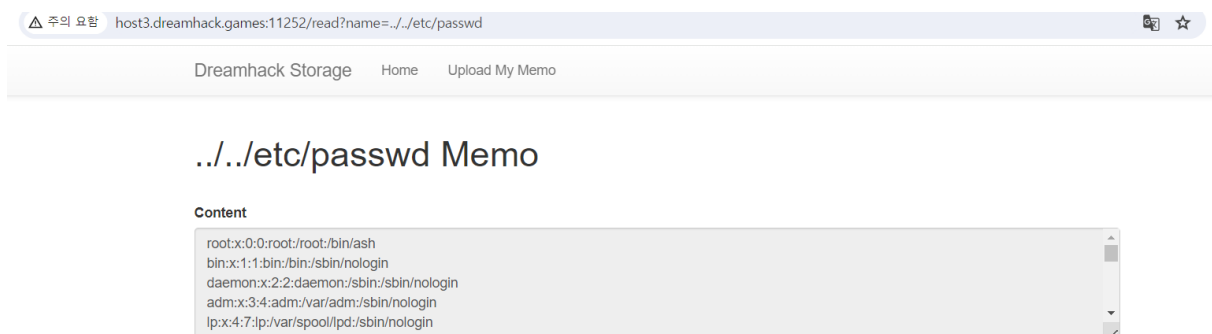
[풀이 작성하고 포인트 받기 >](#)



성공!

나는 파일을 업로드할 때 FLAG 값을 얻을 수 있을 것이라 생각하여 filename에 `../../uploads/flag.py` 나 `../flag.py` 등등 이렇게 저렇게 넣는 시도만 해보았고 URL을 볼 생각을 전혀 하지 못했다.. 다른 도움 없이 풀어보고 싶다.. ㅎ


번외로,,



케칠주에서 배웠던 것을 되살려 `../../etc/passwd`로 변경해보기도 해보았다. 값이 나온다. 신기하다.

### [웹 취약점]파일 다운로드 취약점의 정의 및 대응 방안 ( File Download Vulnerability )

파일 다운로드 취약점이란 ? 웹 서버의 파일 시스템에 존재하는 파일을 다운로드하는 과정 상에서 파일의 경로를 임의로 조작하여 내부의 자료를 다운로드 할 수 있는 취약점으로 보통 파일 경로 및 파일명을 파라미터로 받아 처리를 하는 경우, 적절한 필터링 조치를 하지 않아 조작이 가능한 경우 발생하는 취약점이다. 파일 다운로드

 [https://maker5587.tistory.com/38#google\\_vignette](https://maker5587.tistory.com/38#google_vignette)

