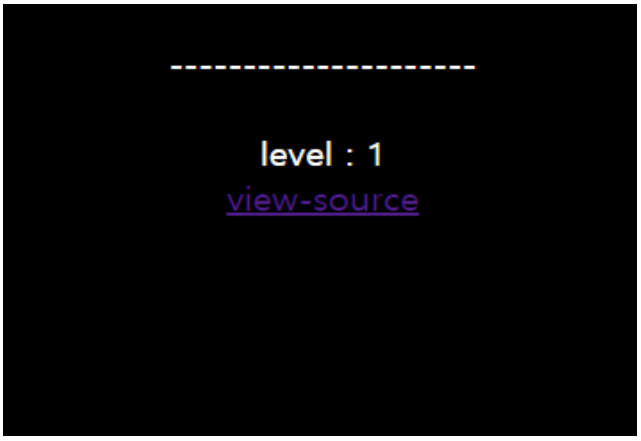
# <웹해킹 3주차 문제풀이>

1,24,(26번은 발표자료에 업로드)>

**1번.**




이번 웹 해킹 1번 문제는 level 1 이라는 글자와 함께 소스코드를 볼 수 있도록 해 놓았다.

```
<?php
  include "../../config.php";
  if($_GET['view-source'] == 1){ view_source(); }
  if(!$_COOKIE['user_lv']){
    SetCookie("user_lv","1",time()+86400*30,"/challenge/web-01/");
    echo("<meta http-equiv=refresh content=0>");
  }
?>
<html>
<head>
<title>Challenge 1</title>
</head>
<body bgcolor=black>
<center>
<br><br><br><br><br>
<font color=white>
----------<br>
<?php
  if(!is_numeric($_COOKIE['user_lv'])) $_COOKIE['user_lv']=1;
  if($_COOKIE['user_lv']>=4) $_COOKIE['user_lv']=1;
  if($_COOKIE['user_lv']>3) solve(1);
  echo "<br>level : {$_COOKIE['user_lv']}";
?>
<br>
<a href=./?view-source=1>view-source</a>
</body>
</html>
```

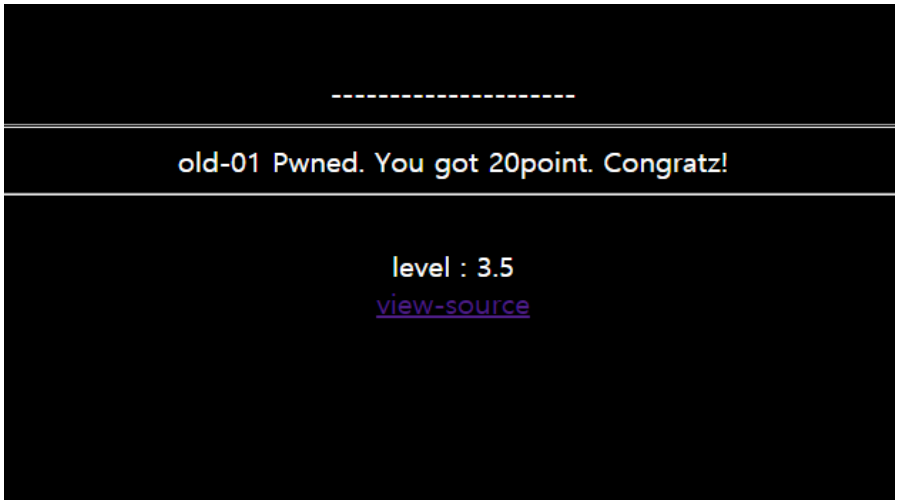
php 구문을 중점으로 확인해보니, is_numeric 함수가 보였다. 이 함수가 무슨 역할을 하는지 잘 몰라서 찾아보았다. Is_numeric 숫자형 데이터 값인 경우 'ture'를 반환하고, 숫자

형 데이터 값이 아닌 경우 'false'를 반환하는 함수이다. 우리가 이번 3주차 강의 내용에서 쿠키를 배웠는데, 아마도 쿠키 값에 숫자를 넣어야지 문제가 풀리지 않을까 라는 생각이 들었다.

```
<?php
  if(!is_numeric($_COOKIE['user_lv'])) $_COOKIE['user_lv']=1;
  if($_COOKIE['user_lv']>=4) $_COOKIE['user_lv']=1;
  if($_COOKIE['user_lv']>3) solve(1);
  echo "<br>level : {$_COOKIE['user_lv']}";
?>
```

php부분을 다시 확대해서 보게 되면, 쿠키 user_lv를 사용하고 user_lv가 없는 경우에는 1로 초기화한다는 것을 알 수 있다. (첫번째 줄 코드) 그리고, 그 다음줄에 user_lv가 3초과 4미만 일 때 문제가 풀리는 것을 알 수 있다.

그러면 우리가 상식적으로 3초과 4미만인 정수는 존재하지 않기 때문에 자연스럽게 실수를 생각해볼 수 있다. 그래서 쿠키의 값을 3.5라고 가정하고 대입을 해보았다.




level 옆에 있는 숫자 값이 3.5로 변하였고, 성공한 것을 알 수 있었다.

(참고로, php구문의 마지막줄을 보면 'level: 쿠키의 user_lv의 값' 을 출력하는 것을 알 수 있다.)

**24번.**

| client ip | 218.156.4.168 |
|---|---|
| agent | Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.82 Safari/537.36 |

Wrong IP!

view-source

이번 문제는 ip주소와 agent를 가져온 페이지가 가장 먼저 떠 있었다. 이번에도 소스 코드를 살펴보니 다음과 같았다.

```
<?php
  include "../../config.php";
  if($_GET['view_source']) view_source();
?><html>
<head>
<title>Challenge 24</title>
</head>
<body>
<p>
<?php
  extract($_SERVER);
  extract($_COOKIE);
  $ip = $REMOTE_ADDR;
  $agent = $HTTP_USER_AGENT;
  if($REMOTE_ADDR){
    $ip = htmlspecialchars($REMOTE_ADDR);
    $ip = str_replace("..",".",$ip);
    $ip = str_replace("12","",$ip);
    $ip = str_replace("7.","",$ip);
    $ip = str_replace("0.","",$ip);
  }
  if($HTTP_USER_AGENT){
    $agent=htmlspecialchars($HTTP_USER_AGENT);
  }
  echo "<table border=1><tr><td>client ip</td><td>{$ip}</td></tr><tr><td>agent</td><td>{$agent}</td></tr></table>";
  if($ip=="127.0.0.1"){
    solve(24);
    exit();
  }
  else{
    echo "<hr><center>Wrong IP!</center>";
  }
?><hr>
<a href=?view_source=1>view-source</a>
</body>
</html>
```

이번에도 php부분을 중점적으로 보았다. 가운데 if문을 보니 REMOTE_ADDR 의 값이 있다면 htmlspecialchars 함수를 통해서 특정 문자들을 치환하여 저장하고 , 더 아래에 있는 if문을 보니 ip주소가 127.0.0.1이면 문제가 풀린다는 것을 유추해 볼 수 있었다.

그래서 그냥 가장 쉬운 방법대로 ip주소를 127.0.0.1을 넣어보았는데 ip주소가 1이 나온 나온 것 보아 이 방법은 아닌 것 같았다.

| client ip | 1 |
| --- | --- |
| agent | Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) |

왜 127.0.0.1의 마지막부분인 1만 나왔을까 생각해보니 12(공백)7.(공백)0.(공백)0.(공백)1인 것으로 보아 1만 남은 것이다. (공백처리는 위에 php구문에 str_replace함수 때문인 것을 알 수 있다.)

따라서, 127..(.)0..(.)0..(.)1 이기 때문에 112277...00...00...1 이 된다는 것을 알 수 있다.

(공백처리 되는 것을 하나씩 생각해보면 된다.)

쿠키 이름을 REMOTE_ADDR 로 설정하고 값을 112277...00...00...1 을 설정한 쿠키를 최종적으로 만들어주면 성공이다.