WhiteHat School 1기

Injection 문제 Write-up

Team: 과부화

Writer: 정진교

문제분석

화이트햇스쿨 정도원 멘토(rubiya)님이 제작하신 사이트인 los.rubiya.kr에서
문제를 하나 골라와봤다.

---

query : **select id from prob_zombie_assassin where id='' and pw=''**

```
<?php
  include "./config.php";
  login_chk();
  $db = dbconnect();
  $_GET['id'] = strrev(addslashes($_GET['id']));
  $_GET['pw'] = strrev(addslashes($_GET['pw']));
  if(preg_match('/prob|_|₩.|₩(₩)/i', $_GET[id])) exit("No Hack ~_~");
  if(preg_match('/prob|_|₩.|₩(₩)/i', $_GET[pw])) exit("No Hack ~_~");
  $query = "select id from prob_zombie_assassin where id='{$_GET[id]}' and pw='{$_GET[pw]}'";
  echo "<hr>query : <strong>{$query}</strong><hr><br>";
  $result = @mysqli_fetch_array(mysqli_query($db,$query));
  if($result['id']) solve("zombie_assassin");
  highlight_file(__FILE__);
?>
```

코드를 살펴보면 입력된 값을 받는 변수는
id와 pw가 있는 것을 확인할 수 있다.

id와 pw 변수에 값을 받는 줄을 살펴보면
addslashes 함수와 strrev 함수를 사용하고 있다.

addslashes 함수의 역할은 싱글 쿼터, 더블 쿼터, 역슬래시, null의 입력값에 말그대로
역슬래시를 추가해 공격 구문을 문자열로 인식하게 만들어준다.

strrev 함수의 역할은 문자열을 거꾸로 반환을 한다.

즉, id로 싱글쿼터를 입력하게되면 addslashes 함수에 의해 \가 입력이 된다.
추가적으로 strrev 함수로 인해 문자열이 거꾸로 반환이 되어 \가 된다.

그렇기에 만약 ?id=’ 구문을 입력하게 되면

select id from prob_zombie_assassin where id=’ ‘\’ and pw=’ ‘

위 쿼리문처럼 되는데 완벽하지 않은 쿼리문으로 에러가 난다.

위에서 말한 것처럼 addslashes 함수는 싱글 쿼터, 더블 쿼터, 역슬래시, NULL에 \ 처리를하기 때문에, ?id=\를 입력하게 된다면

select id from prob_zombie_assassin where id=’ \\’ and pw=’ ‘

위 구문처럼 id의 ’ 를 문자로 인식하게 하지 못한다.

그렇기에 우리는 id의 값으로 “(더블쿼터)나 %00(NULL)값을 입력해주어야 한다.

?id= “ 구문을 입력하게 된다면

select id from prob_zombie_assassin where id=’ “\’ and pw=’ ‘

id의 ’ (싱글쿼터)가 \처리가 되어 문자로 인식이되며 pw의 ‘까지 id의 입력 값으로 인식이 되는 것이다.

php 코드 중

if($result['id']) solve("zombie_assassin"); 부분을 보면

PHP에서는 값이 0이거나 빈 문자열 또는 NULL이 아닌 경우를 참으로 간주한다.

따라서 ’ id’ 키를 가진 값이 존재하고 비어 있지 않으면 조건은 참이 된다.

preg_match 함수로 인해서 url 파라미터 ‘id’에 'prob' 문자열이나 ” (더블 쿼터) 혹은

.(마침표)나 괄호 문자가 입력 되면 사용자의 입력을 거부하기 때문에 해당 구문들을 다른 방식으로 필터링해서 접근해야 한다.

문제풀이

취약점… lssmsmam

id 값에 “\, pw 값에 or 1#값을 넣기 위해

그렇기에 우리는 다음과 같은 구문을 입력할 것이다.

?id="&pw=%231 ro <- 이 구문에서 %23은 #을 url인코딩을 한 것이다.

URL을 조작할 시에는 url 인코딩을 이용한다.

해당 구문을 입력하게 되면

where id=’ “\ <- addslashes 함수로 인해 역슬래시가 붙게 되고

where id=’ “\” & <- &는 and로 인식,

where id=’ “\” &pw=%231 <- %231은 #1 url 인코딩

where id=’ “\” &pw=%231 ro <- ro는 strrev 함수 때문에 거꾸로 작성.

결과

query : **select id from prob_zombie_assassin where id='₩' and pw='or 1#'**

ZOMBIE_ASSASSIN Clear!

```php
<?php
  include "./config.php";
  login_chk();
  $db = dbconnect();
  $_GET['id'] = strrev(addslashes($_GET['id']));
  $_GET['pw'] = strrev(addslashes($_GET['pw']));
  if(preg_match('/prob|_|₩.|₩(₩)/i', $_GET[id])) exit("No Hack ~_~");
  if(preg_match('/prob|_|₩.|₩(₩)/i', $_GET[pw])) exit("No Hack ~_~");
  $query = "select id from prob_zombie_assassin where id='{$_GET[id]}' and pw='{$_GET[pw]}'";
  echo "<hr>query : <strong>{$query}</strong><hr><br>";
  $result = @mysqli_fetch_array(mysqli_query($db,$query));
  if($result['id']) solve("zombie_assassin");
  highlight_file(__FILE__);
?>
```