

XCTF favorite_number wp

原创

[Garybr0](#) 于 2021-01-14 22:14:20 发布 416 收藏 6

分类专栏: [CTF writeup PHP函数](#) 文章标签: [php数组整形溢出](#) [命令行绕正则过滤](#)

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/weixin_45253216/article/details/112637741

版权



[CTF writeup](#) 同时被 2 个专栏收录

16 篇文章 0 订阅

订阅专栏



[PHP函数](#)

4 篇文章 0 订阅

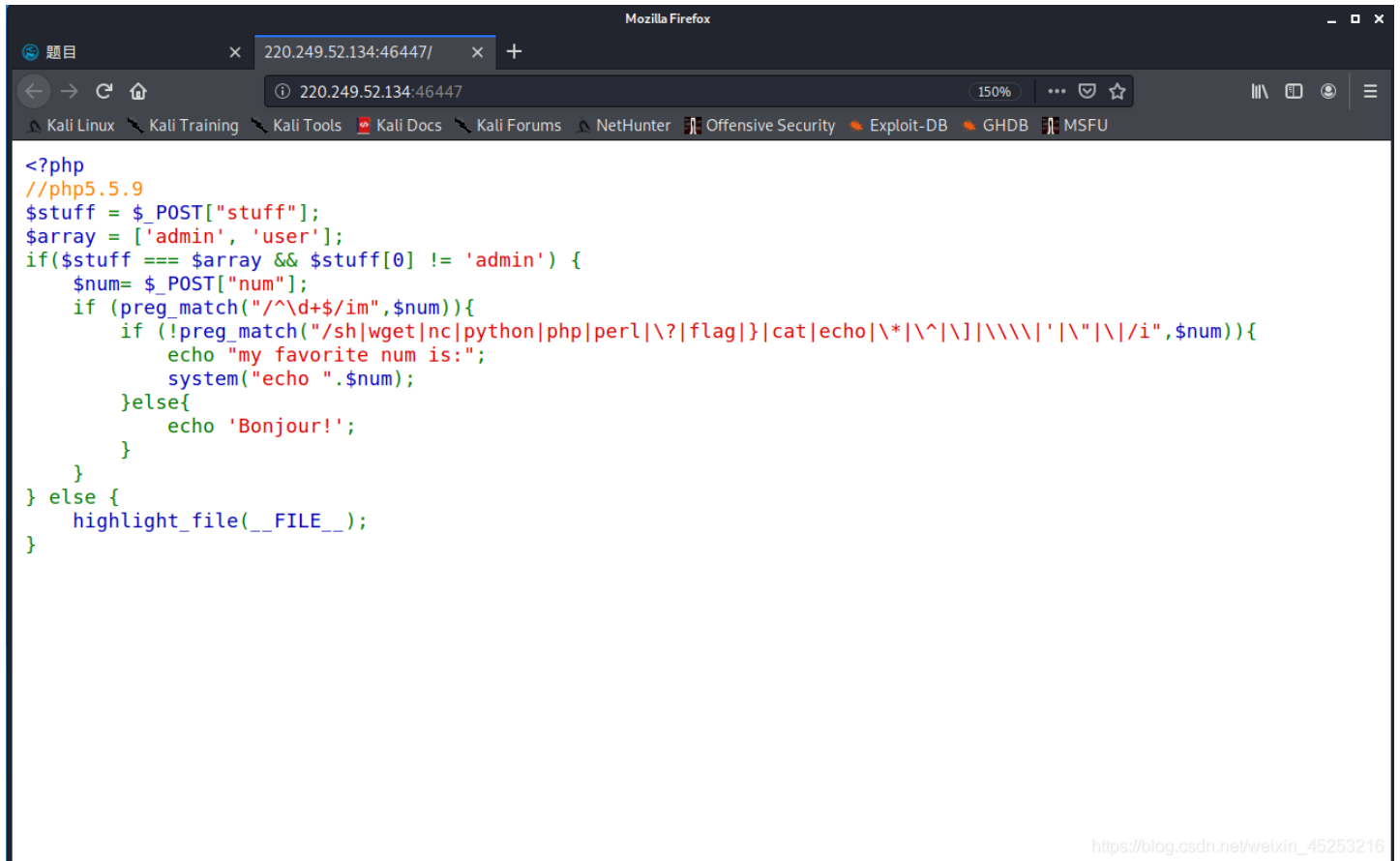
订阅专栏

网安菜鸡今天来划水了!

CTF题目属于萌新入门级, 写下WP仅供自己总结练习, 大佬请自行绕路, 另外如果有师傅愿意有每日轻松一笑环结, 还望不吝赐教。【狗头】

题解:

首先打开题目环境，是一种比较熟悉的格式，页面显示一段php源码：



```
<?php
//php5.5.9
$stuff = $_POST["stuff"];
$array = ['admin', 'user'];
if($stuff === $array && $stuff[0] != 'admin') {
    $num= $_POST["num"];
    if (preg_match("/^\d+$/im",$num)){
        if (!preg_match("/sh|wget|nc|python|php|perl|?|flag|}|cat|echo|\*|\^|\|\\\|\"|'|\|\/i",$num)){
            echo "my favorite num is:";
            system("echo ".$num);
        }else{
            echo 'Bonjour!';
        }
    }
} else {
    highlight_file(__FILE__);
}
```

```
<?php
//php5.5.9
$stuff = $_POST["stuff"];
$array = ['admin', 'user'];
if($stuff === $array && $stuff[0] != 'admin') {
    $num= $_POST["num"];
    if (preg_match("/^\d+$/im",$num)){
        if (!preg_match("/sh|wget|nc|python|php|perl|?|flag|}|cat|echo|\*|\^|\|\\\|\"|'|\|\/i",$num)){
            echo "my favorite num is:";
            system("echo ".$num);
        }else{
            echo 'Bonjour!';
        }
    }
} else {
    highlight_file(__FILE__);
}
```

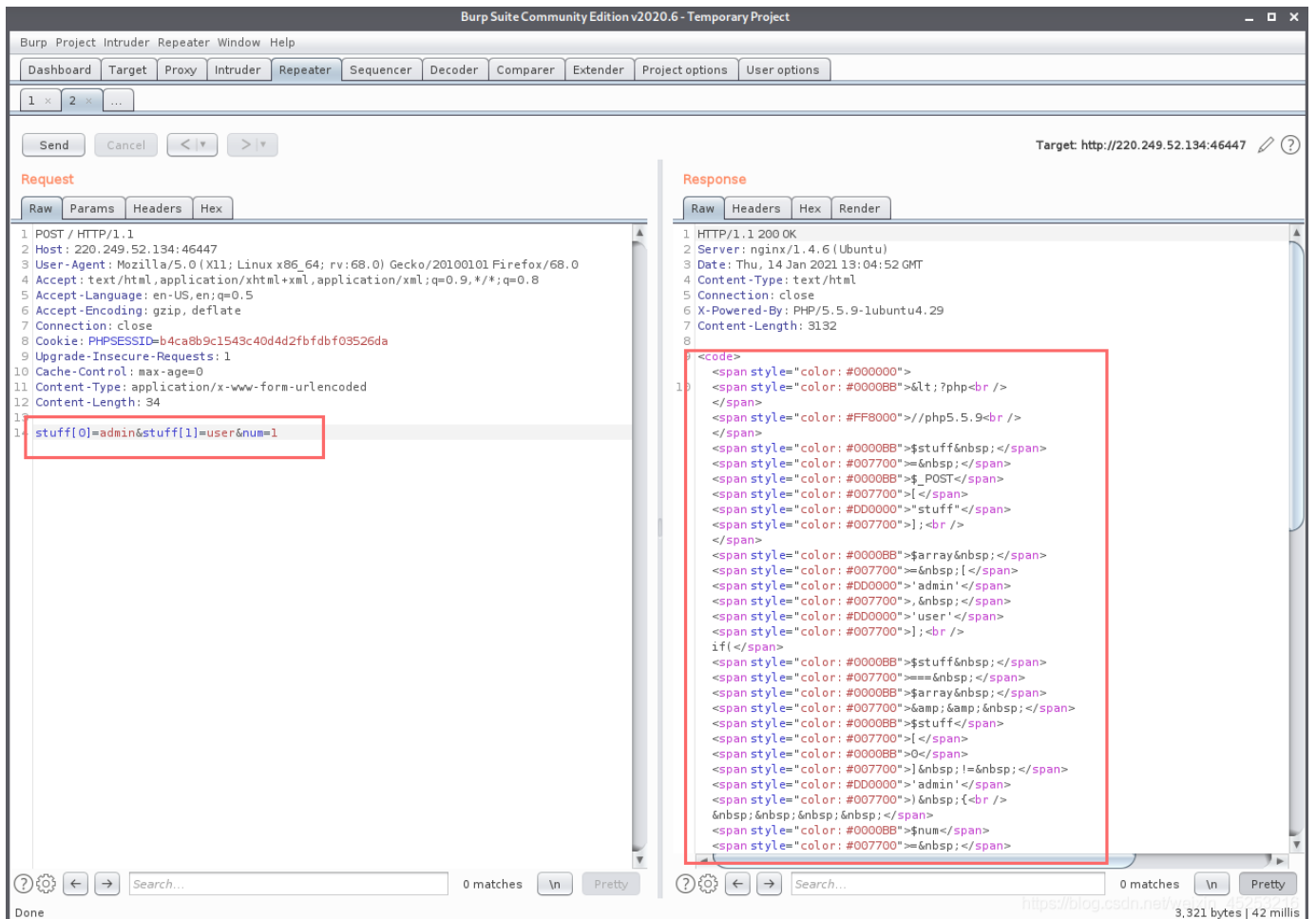
代码审计

首先我们知道是通过POST方式传入stuff这个参数，定义了一个数组变量array，有admin和user。只有当stuff和array这两个数组完全相等（因为使用全等号===）并且stuff索引为0的元素不等于admin，才能往下执行，否则就highlight(__FILE__)。然后通过POST方式传入一个num，然后通过正则过滤（后续详细介绍正则表达式）返回我最喜欢的数字。简要介绍一下本题正则，\d表示必须匹配数字，第二层if循环表示，不能是那些格式的文件，并且过滤了一些特殊字符。

总结一下就是：

1. 首先是个判断，既要数组强等于，又要首元素不等
2. 然后是个正则，要求整个字符串都是数字，大小写不敏感，跨行检测

3. 最后是个黑名单，把常用的都排除了



这里根据题目思路构造payload，这里涉及到第一个知识点：**PHP数组key溢出**，简单的说就是`stuff[4294967296]`表示的**值**，与`stuff[0]`是一个。

这里参考<https://bugs.php.net/bug.php?id=69892>的提示

Bug #69892 Different arrays compare identical due to integer key truncation

Submitted: 2015-06-20 14:29 UTC

Modified: 2015-06-20 14:29 UTC

From: niki@php.net

Assigned: niki (profile)

Status: Closed

Package: Scripting_Engine_problem

PHP Version: 5.5.26

OS:

Private report: No

CVE-ID: None

View

Add Comment

Developer

Edit

[2015-06-20 14:29 UTC] niki@php.net

Description:

```
var_dump([0 => 0] === [0x100000000 => 0]); // bool(true)
```

on all versions: <http://3v4l.org/Sjdf8>

意思是在数组中，这个十六进制数0x100000000,可以当0用。但是在POST传参过程中要转换为十进制。

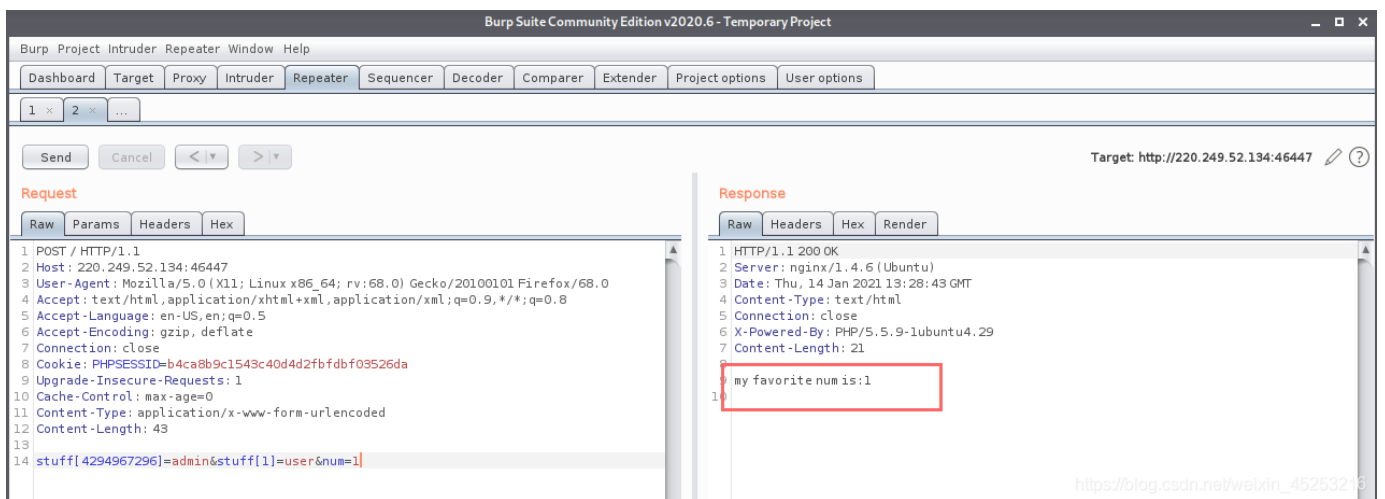
2进制 8进制 10进制 16进制 32进制 64进制 | 更多进制: 16 ▾

步骤：上面选择当前进制，然后下面输入数值，再点【转换】按钮，就能得到常见的进制数据。

进制	结果
二进制	10000000000000000000000000000000
四进制	100000000000000000
八进制	4000000000
十进制	4294967296
十六进制	10000000
三十二进制	4000000
六十四进制	EAAAAA

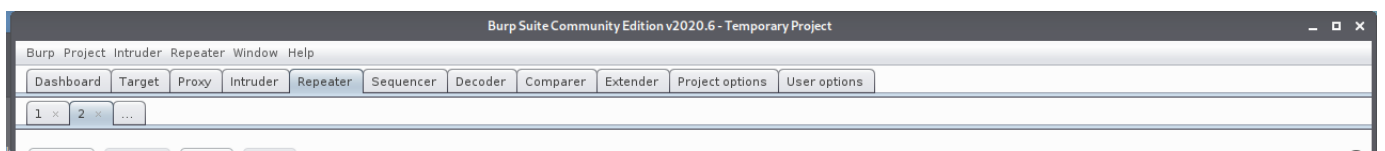
https://blog.csdn.net/weixin_45253216

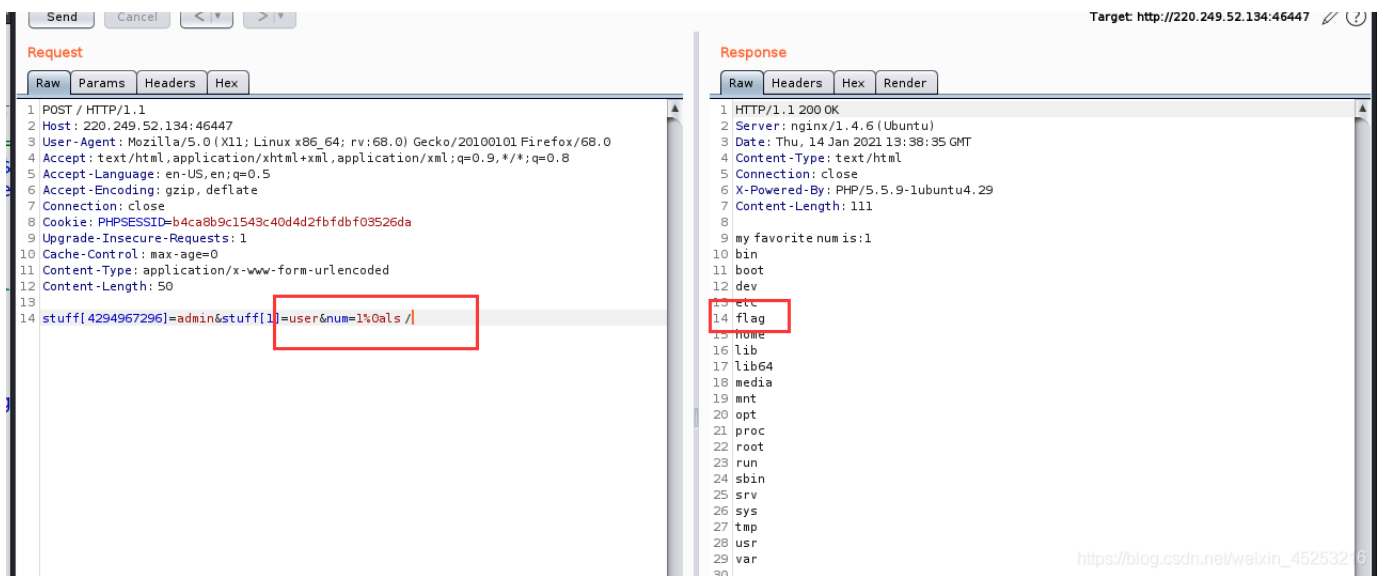
所以4294967296就是这么来的。



https://blog.csdn.net/weixin_45253216

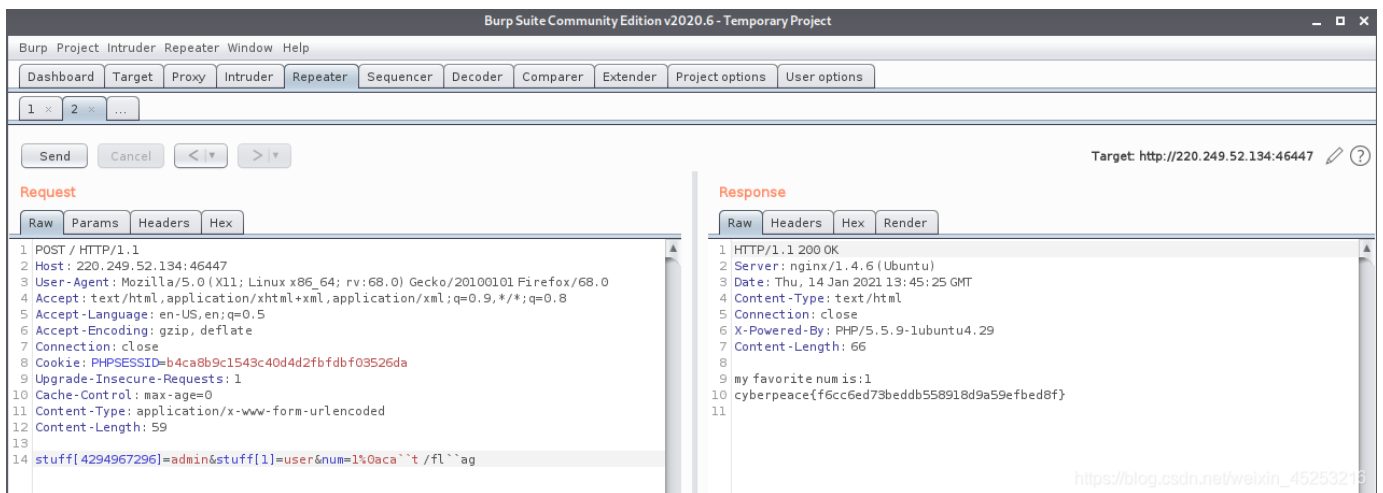
因为preg_match()开启了/m，也就是开启了多行匹配，因此^和\$不仅匹配字符串的开头和结尾，也能匹配一行的开头和结尾，因此可以利用%0a换一行，把命令写在其他的行，这样这个正则匹配就只能匹配到第一行了。





发现flag，这里其实把sh wget nc python...这些过滤掉，就是过滤掉了反弹shell，所以要换姿势绕过。

一 使用反引号绕过



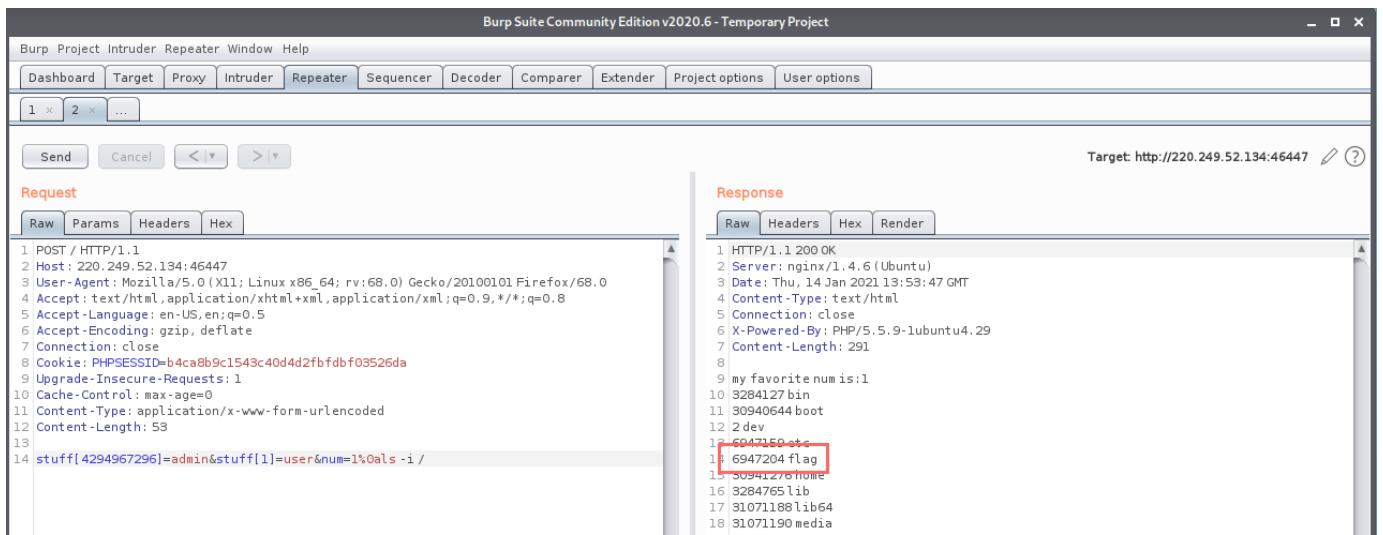
正常可以使用 `ca't ca"'t ca/t`

但是这题都给过滤掉了。

二 使用文件inode

显示命令ls有几个参数-a显示全部文件包括隐藏文件，-l长文件格式即显示全部信息，-h已合适单位显示文件大小，-d只显示目录文件，-t按时间显示，-i查看文件的inode号，inode存储文件的详细信息。

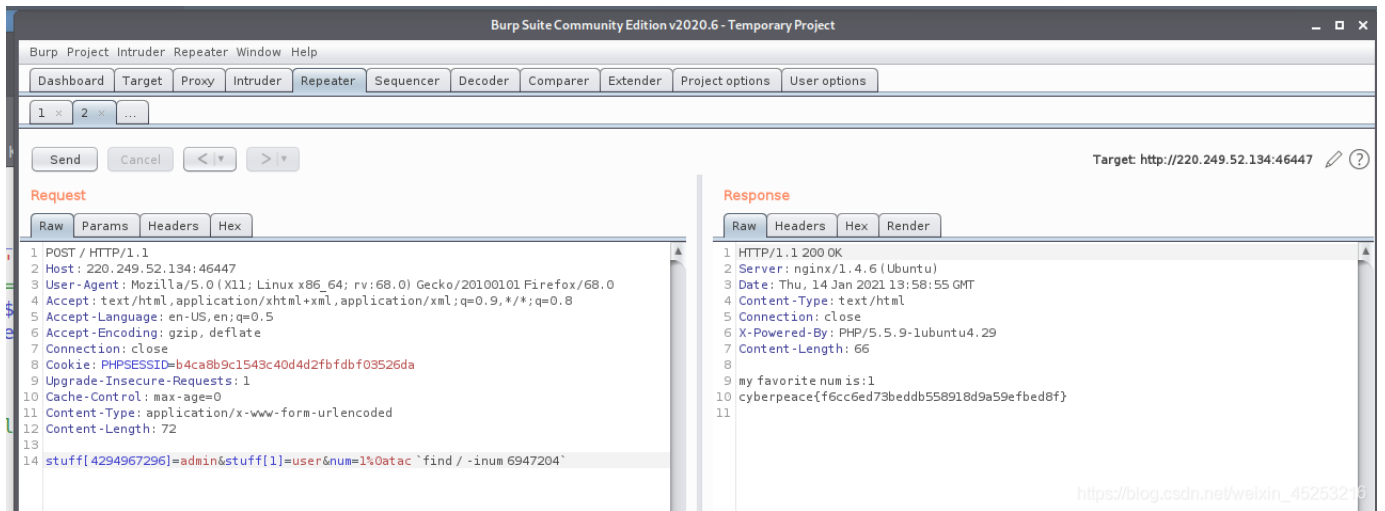
查看flag的文件号。



```
20 31071192 opt
21 1 proc
22 31071194 root
23 31466142 run
24 31466109 sbin
```

https://blog.csdn.net/weixin_45253218

因为cat被过滤了使用tac查看，利用反引号读取文件



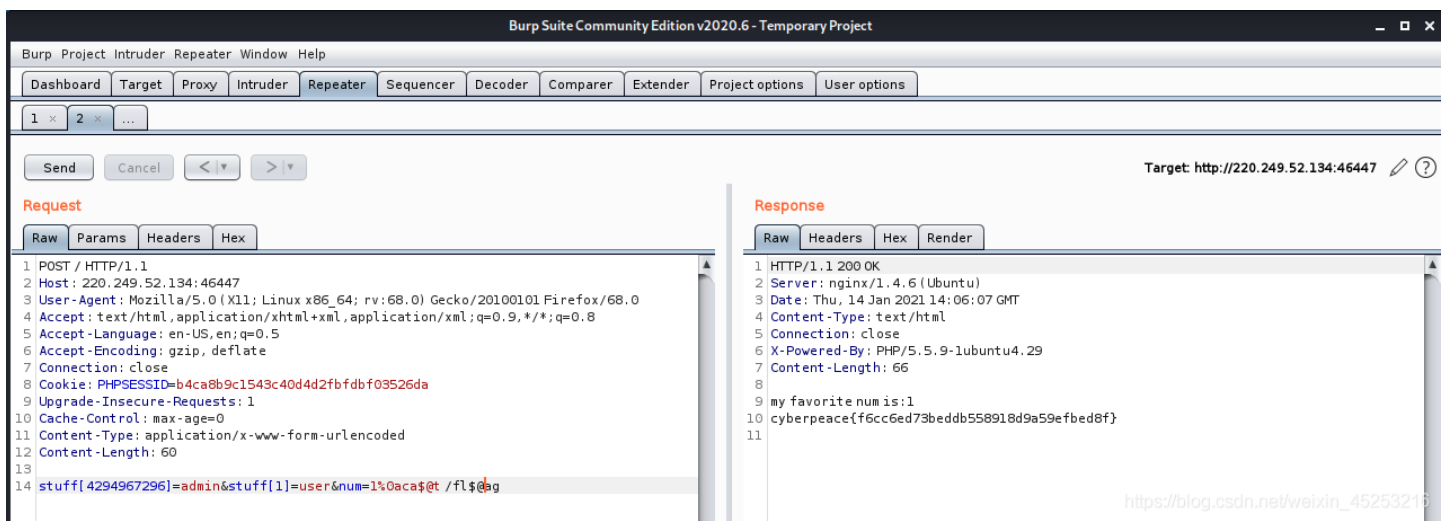
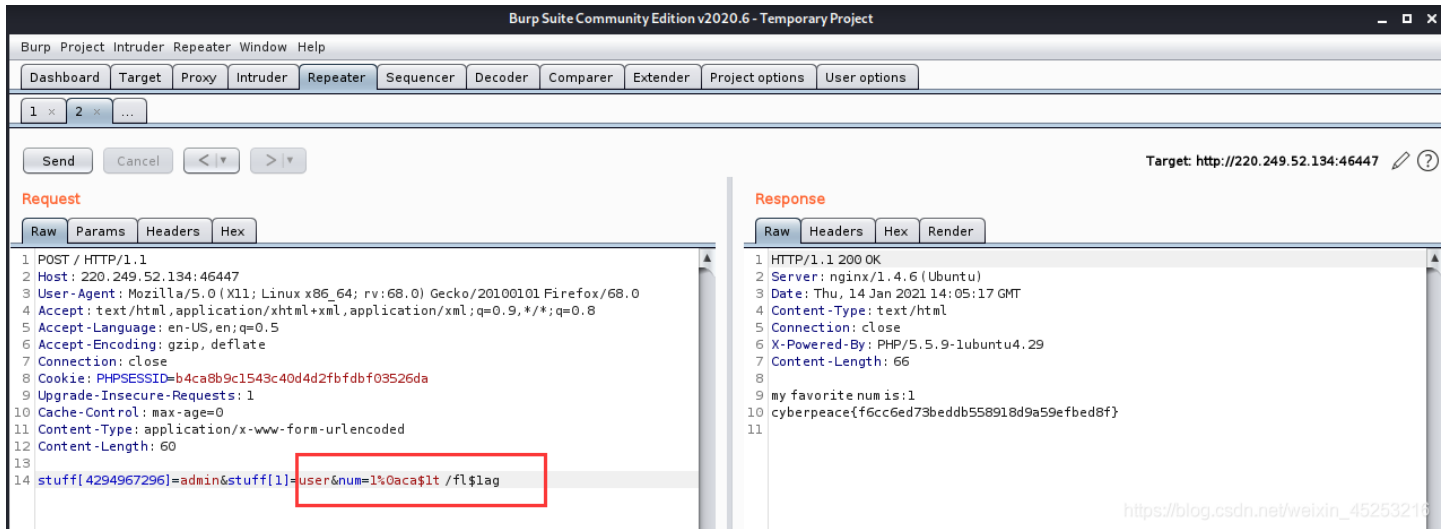
https://blog.csdn.net/weixin_45253218

参考大佬的wp，get了这几个方法，都记录下来。

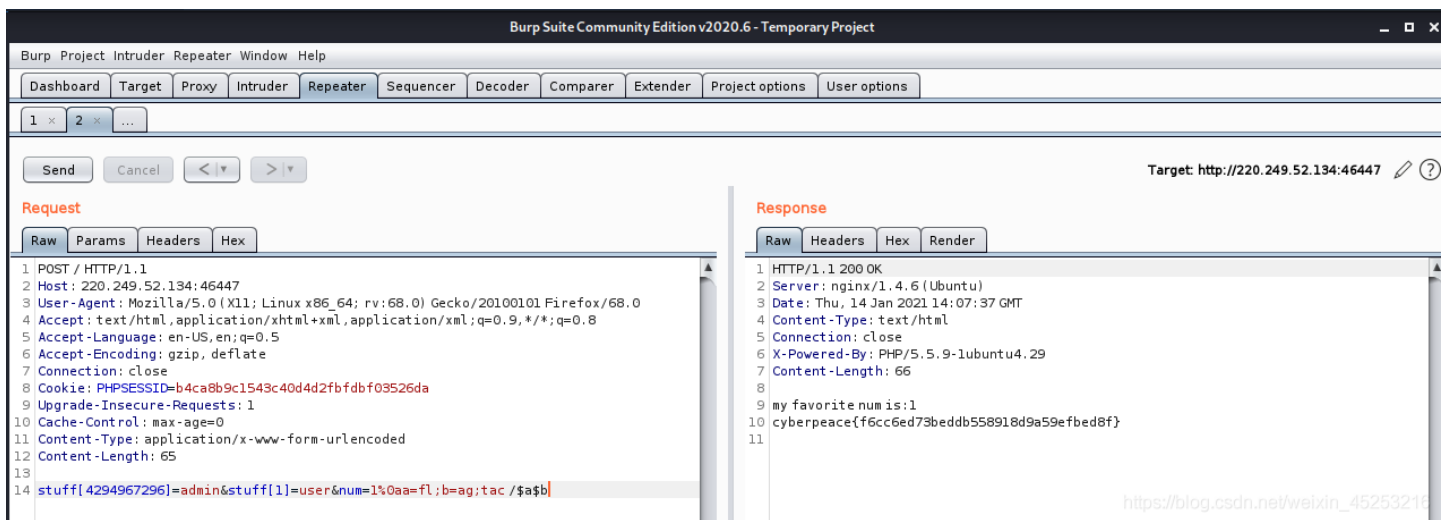
利用\$1 和 \$@

*和@,

$x(x \text{ 代表 } 1 - 9)$, $\{x\}(x \geq 10)$: 比如 `ca${21}t a.txt` 表示 `cat a.txt` 在没有传入参数的情况下, 这些特殊字符默认为空



拼接变量



总结一下:

- PHP5的数组整形溢出
- PHP的preg_match()的/m绕过
- cat等命令行的绕过姿势

关于正则表达式:

简写	描述
.	除换行符外的所有字符
\w	匹配所有字母数字, 等同于 <code>[a-zA-Z0-9_]</code>
\W	匹配所有非字母数字, 即符号, 等同于: <code>[^\w]</code>
\d	匹配数字: <code>[0-9]</code>
\D	匹配非数字: <code>[^\d]</code>
\s	匹配所有空格字符, 等同于: <code>[\t\n\f\r\p{Z}]</code>
\S	匹配所有非空格字符: <code>[^\s]</code>
\f	匹配一个换页符
\n	匹配一个换行符
\r	匹配一个回车符
\t	匹配一个制表符
\v	匹配一个垂直制表符
\p	匹配 CR/LF (等同于 <code>\r\n</code>), 用来匹配 DOS 行终止符

标志也叫模式修正符, 因为它可以用来修改表达式的搜索结果. 这些标志可以任意的组合使用, 它也是整个正则表达式的一部分.

标志	描述
i	忽略大小写.
g	全局搜索.
m	多行的: 锚点元字符 <code>^</code> <code>\$</code> 工作范围在每行的起始.



[创作打卡挑战赛](#) >

[赢取流量/现金/CSDN周边激励大奖](#)