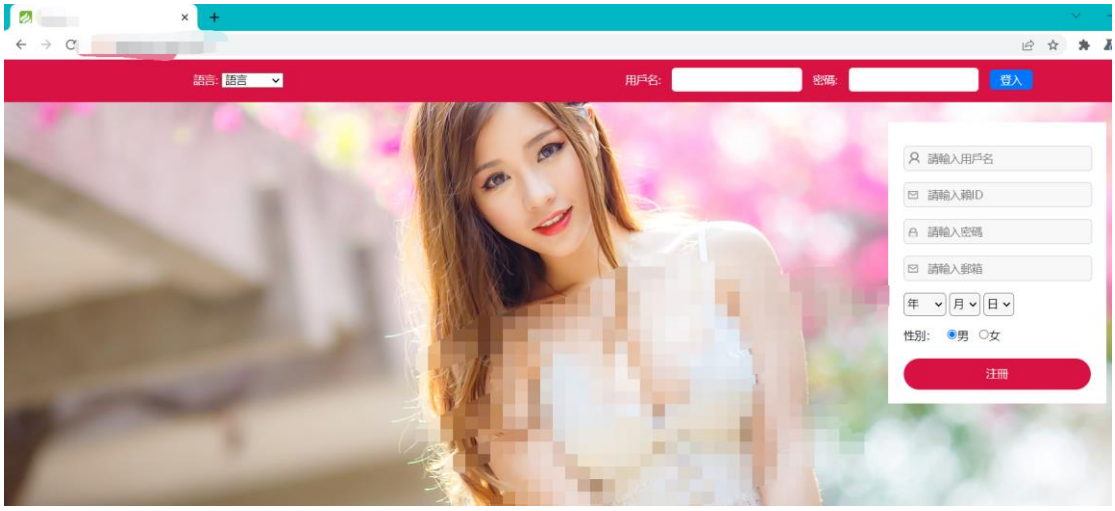
打开一个站点
懂得都懂是什么站，打开后



常规思路:

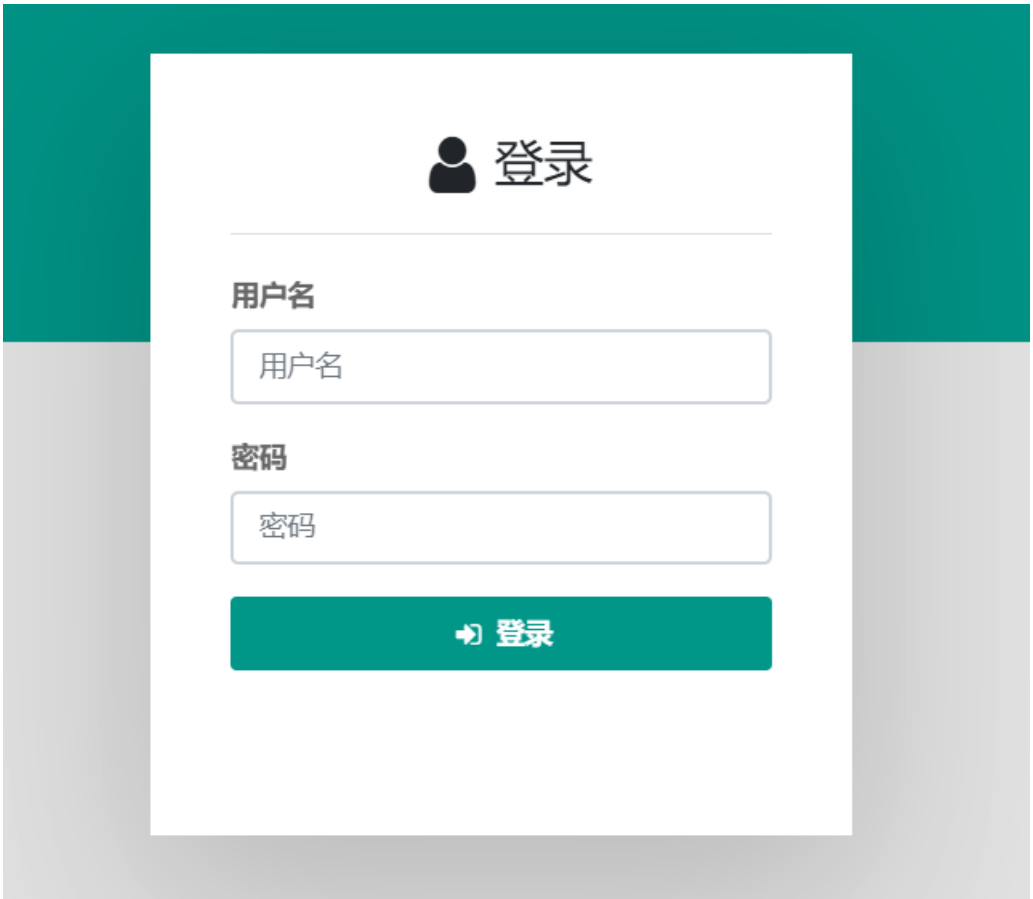
目录扫描

找 IP

找子域名

这里通过 360 威胁情报找到一个子域名

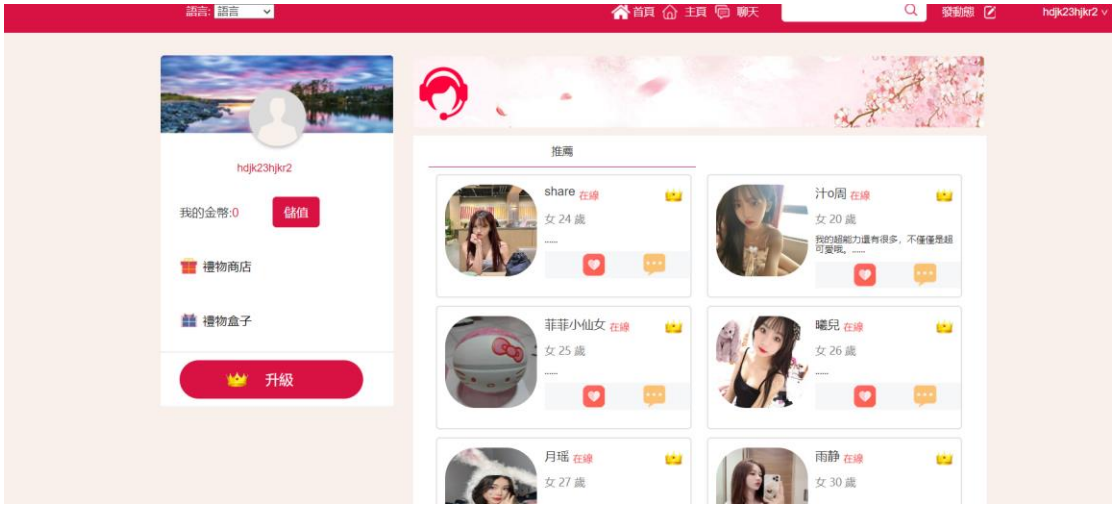
admin.xxxx.com



应该是管理员后台管理的也是 thinkphp 的图标，但是利用工具发现无效后

尝试对登录框进行注入，发现无果，

看到是一个 thinkphp 的 ico，使用 thinkphp 利用工具发现无用，那就注册进去看看有什么东西。
这个注册框有一个 xss 漏洞，但是我不是利用这个漏洞打进去，是用的别的。
发现是一个这个网站



发现资料设置这里可以上传图片

```
POST /index/jpgsave HTTP/1.1
Host:
Cookie:
Content-Le
Sec-Ch-Ua: "
"Chromium";v="105"
Dnt: 1
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Accept: application/json, text/javascript, */*; q=0.01
X-Requested-With: XMLHttpRequest
Sec-Ch-Ua-Platf
Orig
Sec-Fe
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer:
Accept-Enco
Accept-Language: zh,en;q=0.9,en-CN;q=0.8,zh-CN;q=0.7
Connection: close

base64=
data%3Aimage%2Fjpeg%3Bbase64%2C%2F9j%2F4AAQSkZJRgABAQAAAQABAAD%2F4gIoSUND
X1BST0ZJTEUAAQEAAAIYAAAAAAQwAABtbnRyUkdCIFhZWiAAAAAAAAAAAAAAAABhY3NwAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AABRnWFlaAAABeAAAABRiWFlaAAABjAAAABRyVFJDAAABoAAAAChnVFJDAAABoAAAAChiVFJD
AAABoAAAACh3dHB0AAAByAAAABRjcHJ0AAAB3AAAADxtbHVjAAAAAAAAAAEAAAAMZW5VUwAAA
FgAAAAcAHMAUgBHAEIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

发现上传的数据包是一个 base64 加密的图片

返回包

返回包得到一个图片地址

这个时候观察

Base 加密的数据包

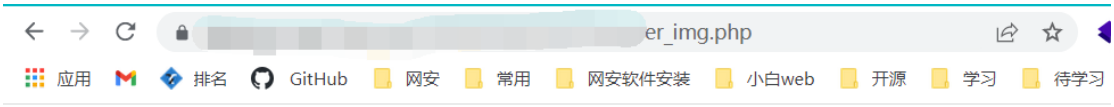
data%3Aimage%2Fjpeg%3Bbase64%2C%2F9j%2F4AAQSkZJRgABAQAAAQABAAD%2F4gIoSUN
D

解码后是

发现前面有一个 jpeg，这个时候可以把这个修改成 php，看看能不能上传

发现成功上传，那么我们上传一个 base 加密的 php 一句话后门看看能不能成功

然后替换到原来的数据包后面



代码审计

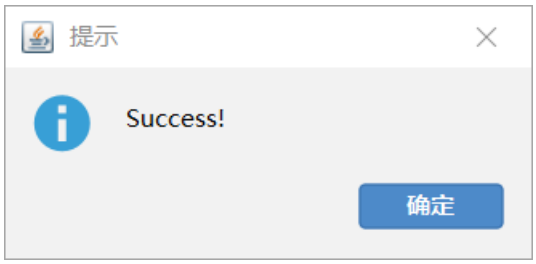
发现$type=='jpeg'时候，后缀就是 jpg，当我们自定义后缀的时候，就不到$type=='jpeg',这个里面，会直接跳到下面的语句，然后图片解密，直接拿 shell

这里发现这个开发这个网站的人并没有使用 thinkphp 官方的开发格式去开发，所以导致的漏洞

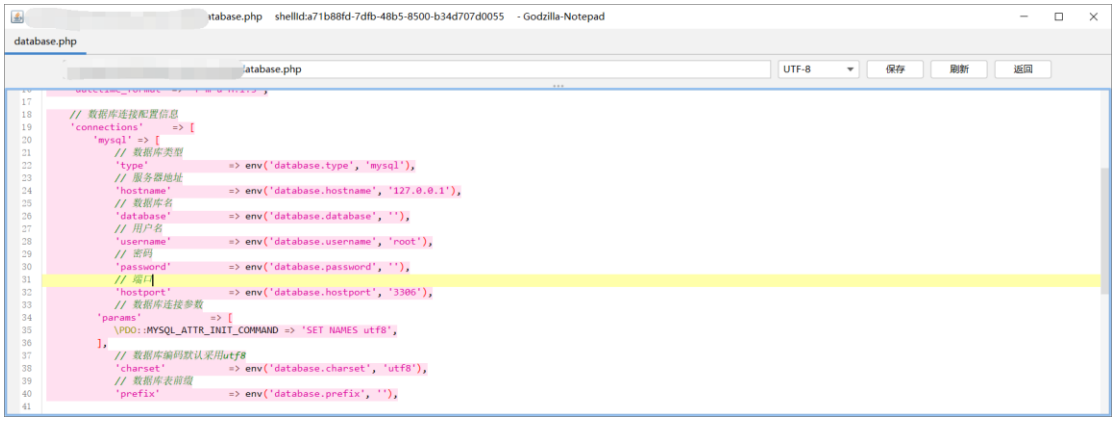
```
1.  /**
2.   * 验证上传的图片
3.   */
4.
5.  function validateImg($base64_image_content){
```

```
6.
7.     if (preg_match('/^(data:\s*image\/(\w+);base64,)/', $base64_image_content, $result)){
8.
9.         //图片后缀
10.        $type = $result[2];
11.        if($type=='jpeg'){
12.            $type='jpg';
13.        }
14.
15.        //解码
16.        $decode=base64_decode(str_replace($result[1], '', $base64_image_content));
17.        $data['code'] = 0;
18.        $data['file_stream'] = $decode;
19.        $data['type'] = $type;
20.        $data['msg'] = '验证成功! ';
21.
22.    }else{
23.        $data['code'] = 1;
24.        $data['msg'] = 'base64 图片格式有误! ';
25.
26.    }
27.    return $data;
28.
29.
30. }
```

得到 shell

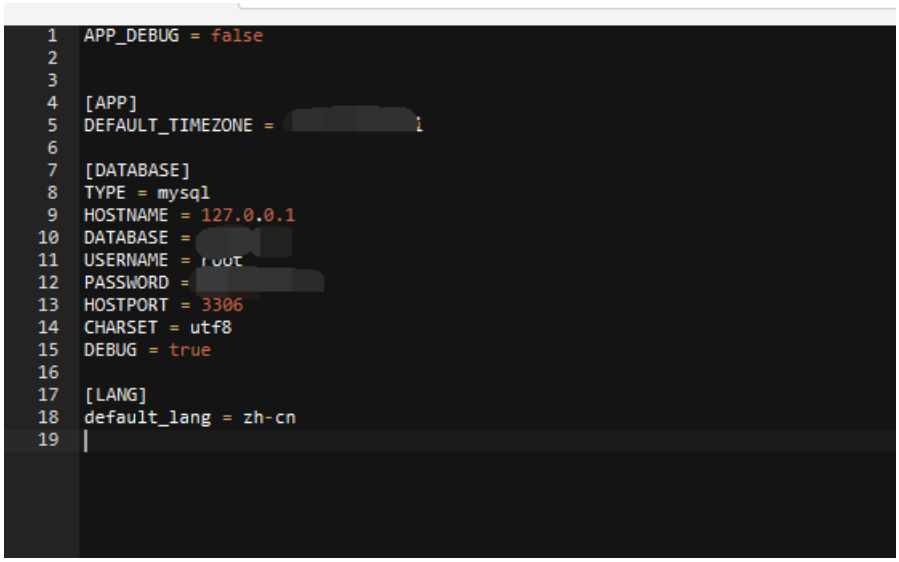


连接成功，然后开始找数据库密码，连接数据库，访问后台管理的用户名和密码发现为空、

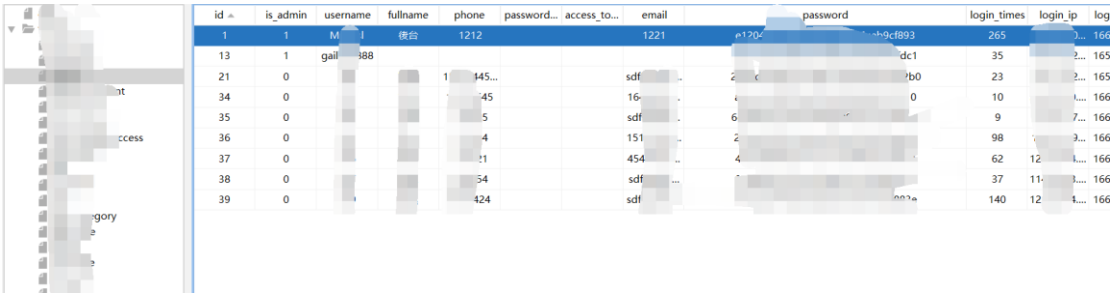


但是这里出现了一个问题，哥斯拉输入空密码的话，连接会失效，所以我这里使用冰蝎

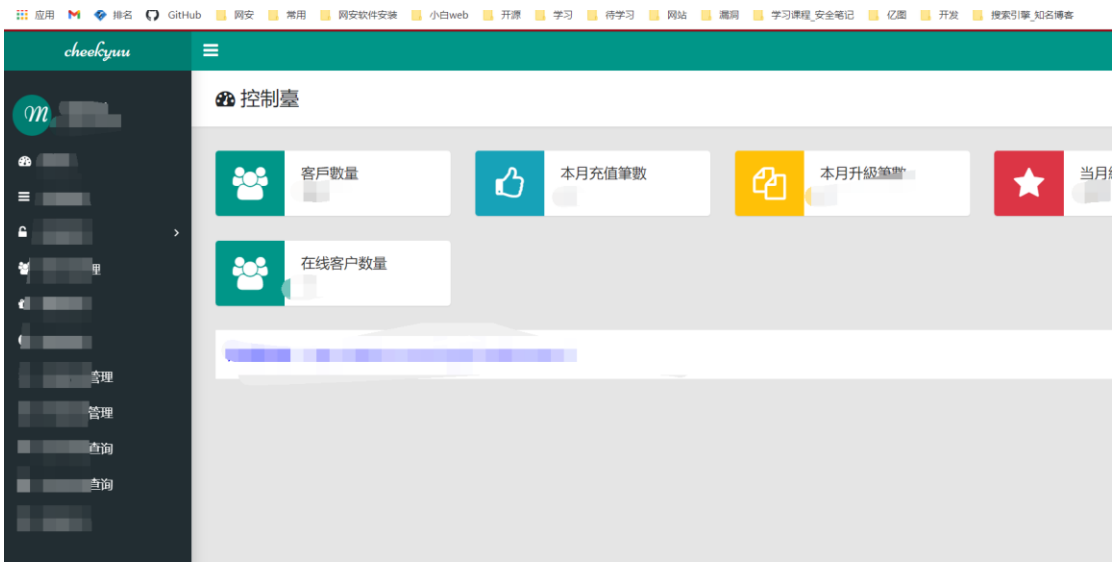
但是，这个是错误的，也就是这个密码是错误的，然后又在其他文件找到了数据库密码



找到管理员账号



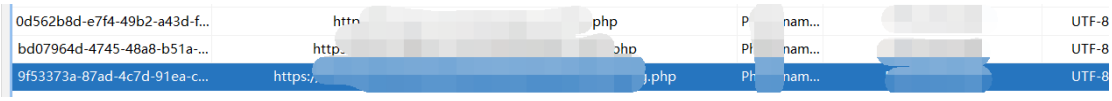
密码为 md5 加密，解密后



登录成功

使用反弹 shell，发现失败，尝试权限提升
漏洞扫描出
CVE-2021-22555
CVE-2021-27365
尝试后失败
然后，尝试使用数据库来执行系统命令，失败。

我在想有没有一样的站点，然后我就去 fofa 搜索指纹，果然，发现了两个也是一样的站点



最后拿到三个 shell