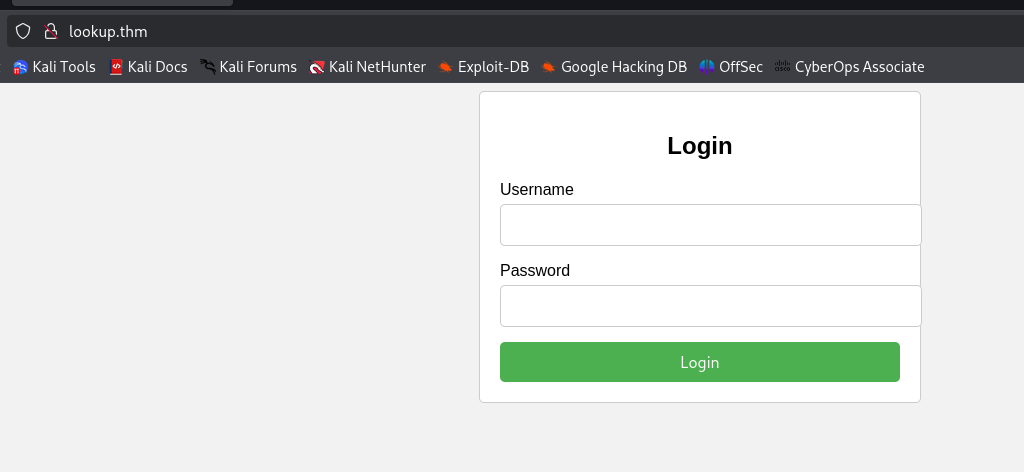
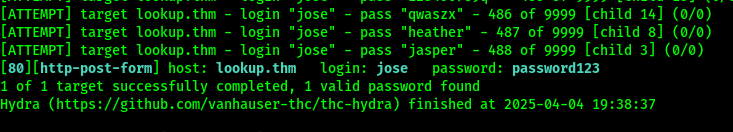
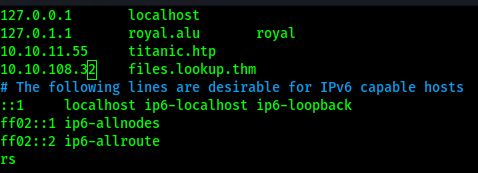
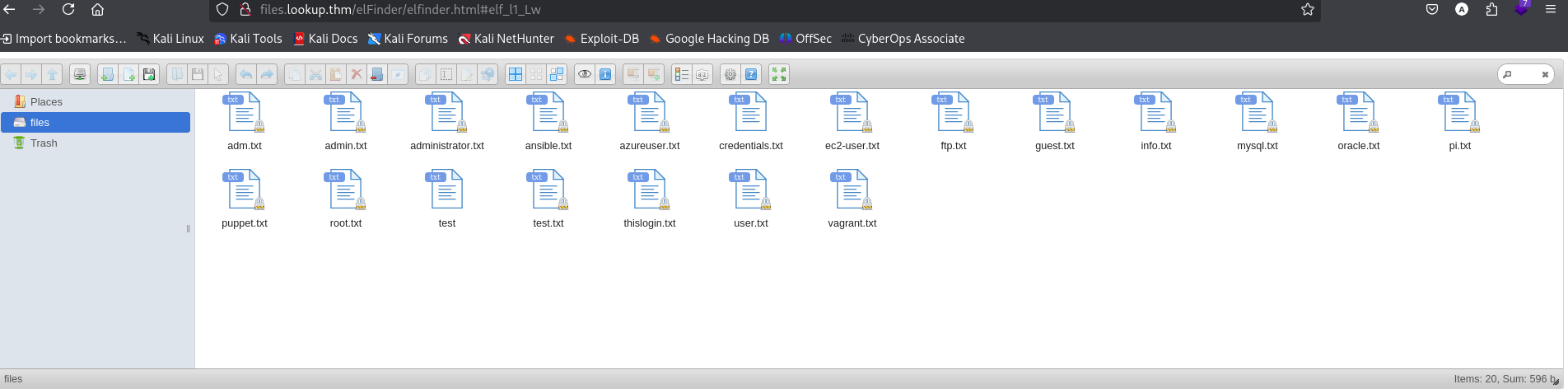
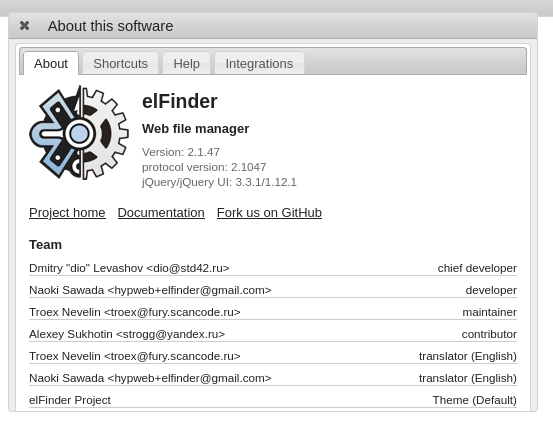
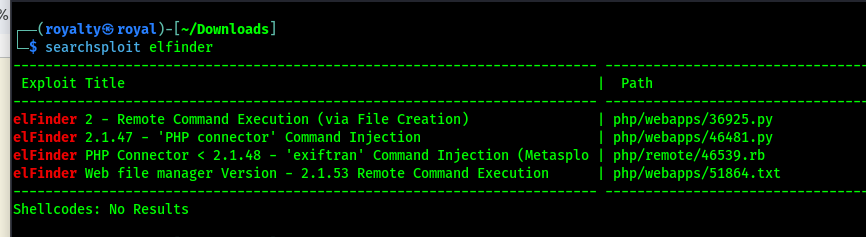
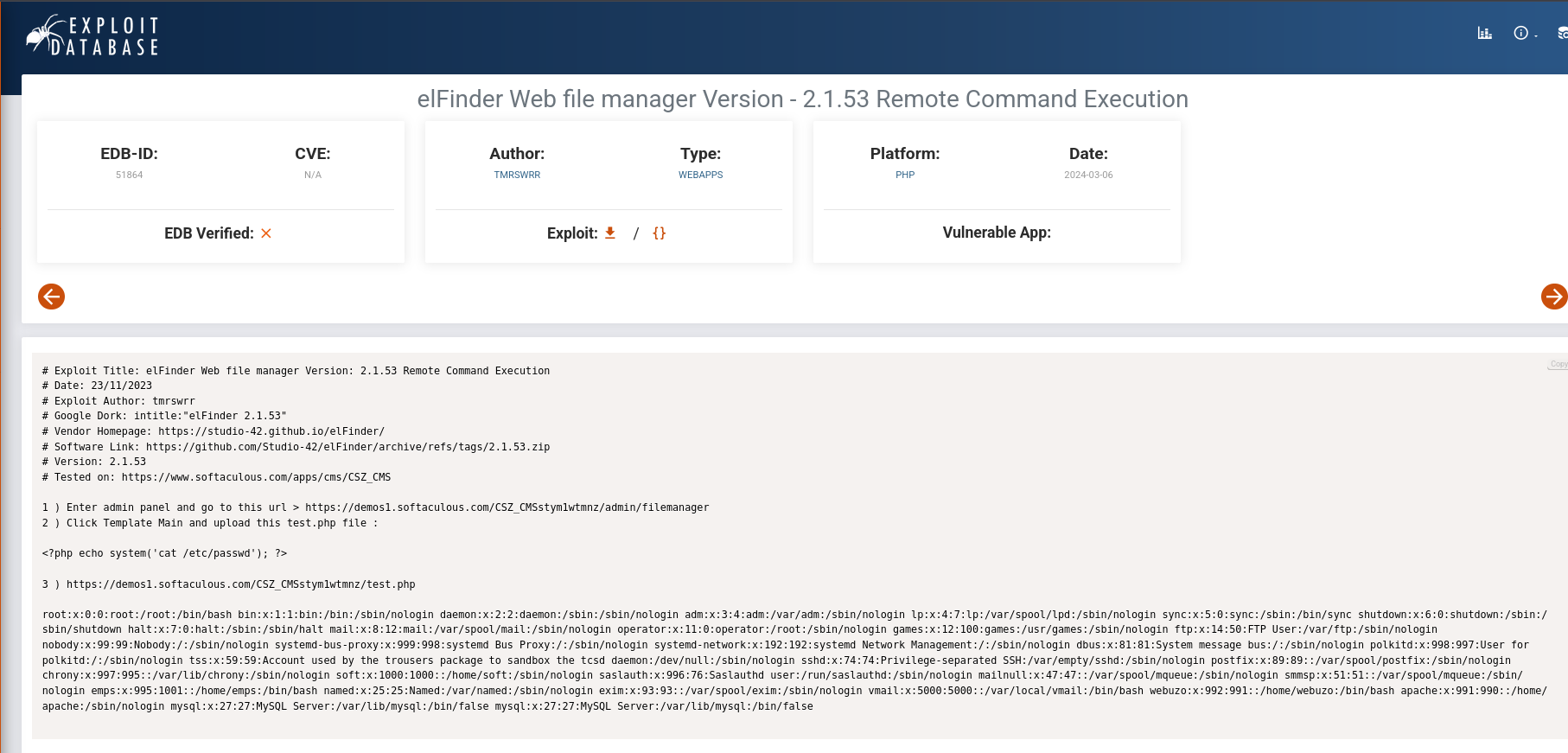
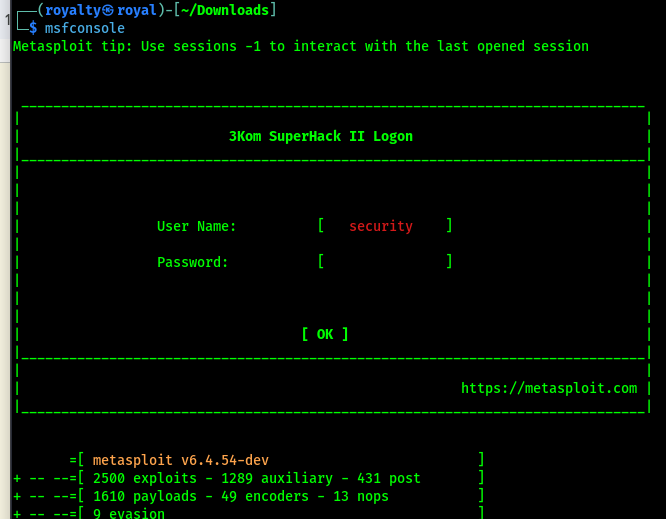
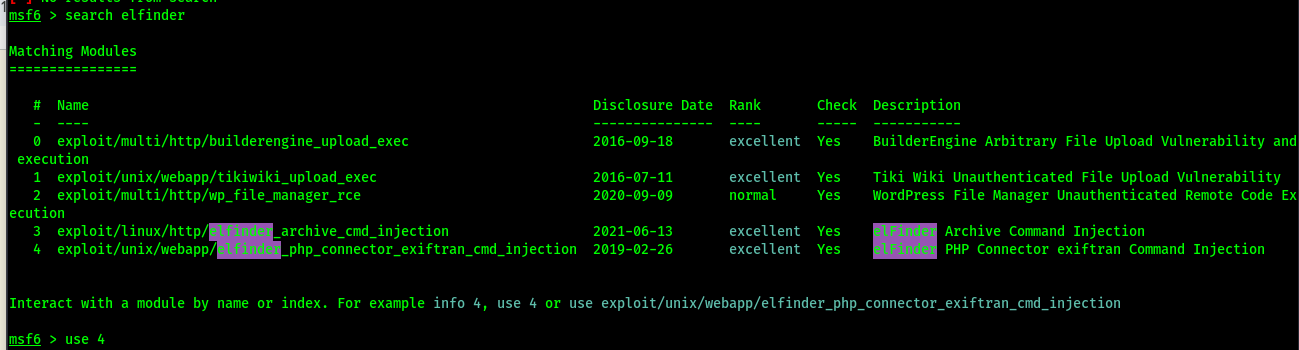
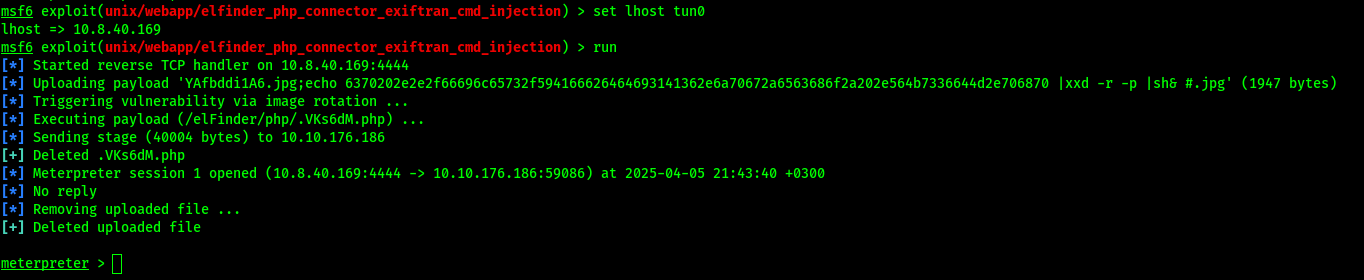
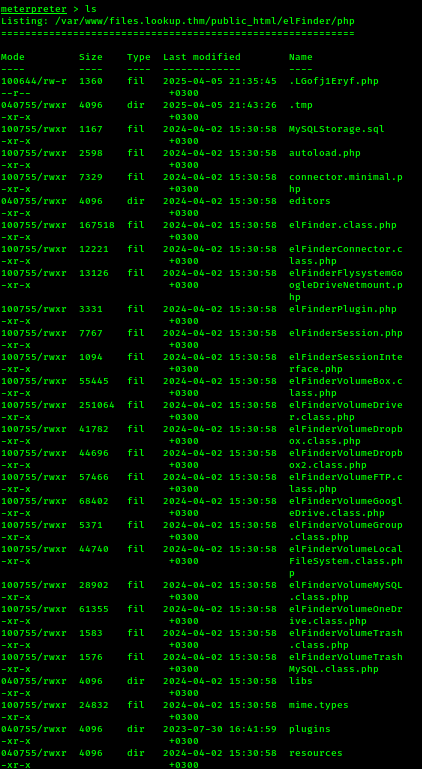
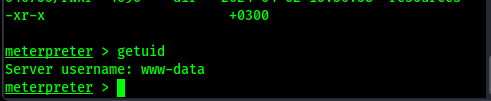
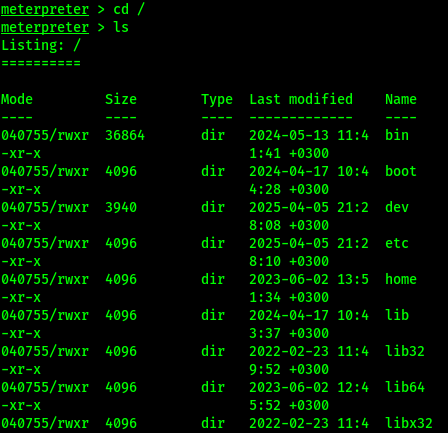
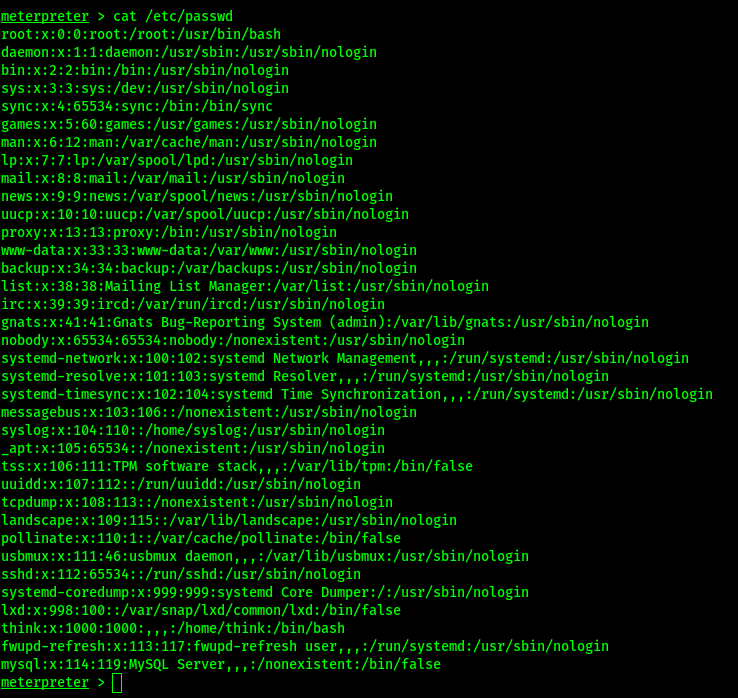
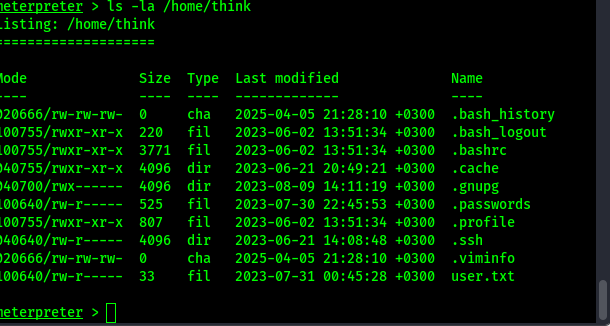
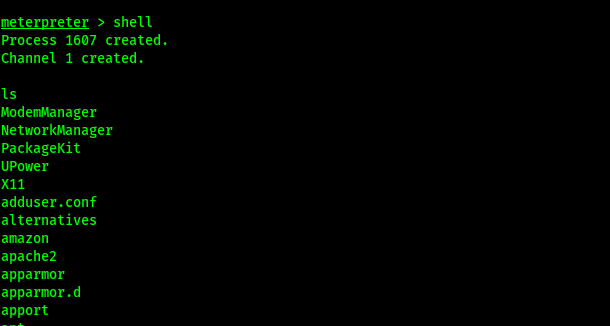
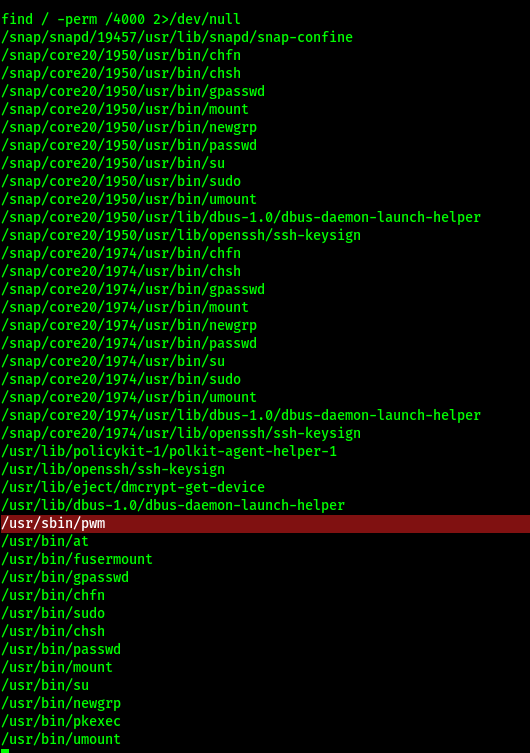
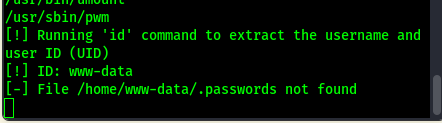
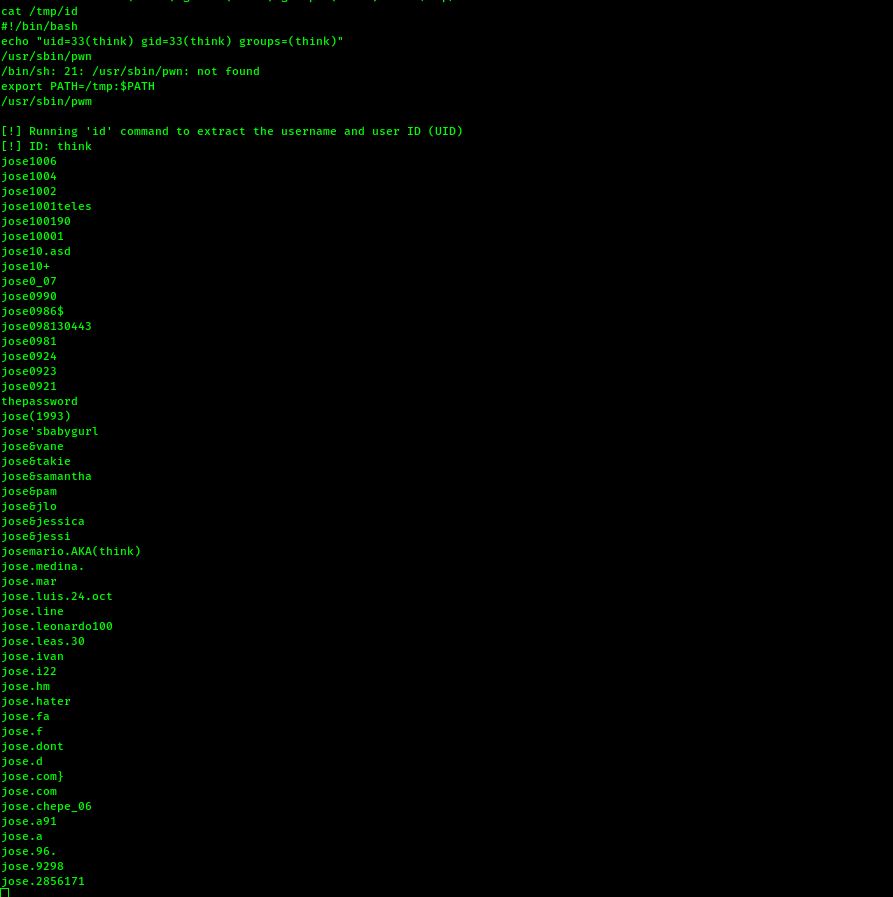
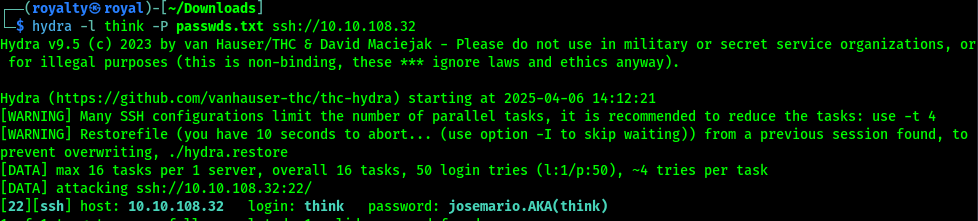
# 3. EXPLOITATION

* I copied the target machine’s ip address and pasted it on my browser. This brings up the login page.
* Now this is obviously a http server as it isn’t secured.
* There are several tools to brute force and find the valid usernames and passwords.
* Now since using hydra and burp suite to brute force will take a longer time, i wrote a python script which ran through a list of names to find a username in the website.
* The script found ‘jose’ as a valid username.
* Now let's use hydra to find a valid password for the username ‘jose’
* Now with ‘jose’ and ‘password123’ as username and password, we can log in to the website.
* Lets change in /etc/hosts from ‘lookup.thm’ to ‘files.lookup.thm’
* This lands us in something that looks like a file manager called elFinder.
* When we look at the about, we find the version of elfinder
* Now we can either use searchsploit or exploit-db to find out more about this version of elFinder



## Privilege escalation

* Lets run metasploit since it is a module targeting version 2.1.47
* Search for elfinder and lets use the 4th option as it matches our previous search using searchsploit
* I set my listening host to my machine as ‘tun0’ and RPORT as ‘files.lookup.thm 'and after running the exploit it opened a meterpreter session.
* Boom we are in. I listed all files in the machine
* Lets get our user id using ‘getuid’ command
* Lets cd to the root folder
* To find other users lets cat the /etc/passwd
* We find ‘think’ as a user and also ‘root’
* Let’s take a look at think’s home directory
* There is a passwords file here but we cannot read it.
* Lets run the shell command to start the shell:
* We have to look for Set owner User ID binaries.
* Looking at this, /usr/bin/pwn looks odd. Lets run it and see the outcome.
* When you run with the -la command you notice it is owned by root.
* So the shell just runs the id command.
* I tried a couple of commands to run the file but came out with errors
* After much trial, i finally got the command to work
* It extracted think as the user id and listed out all possible passwords.
* copy these to our machine and try to brute force the pass for ‘think’



We got the password.