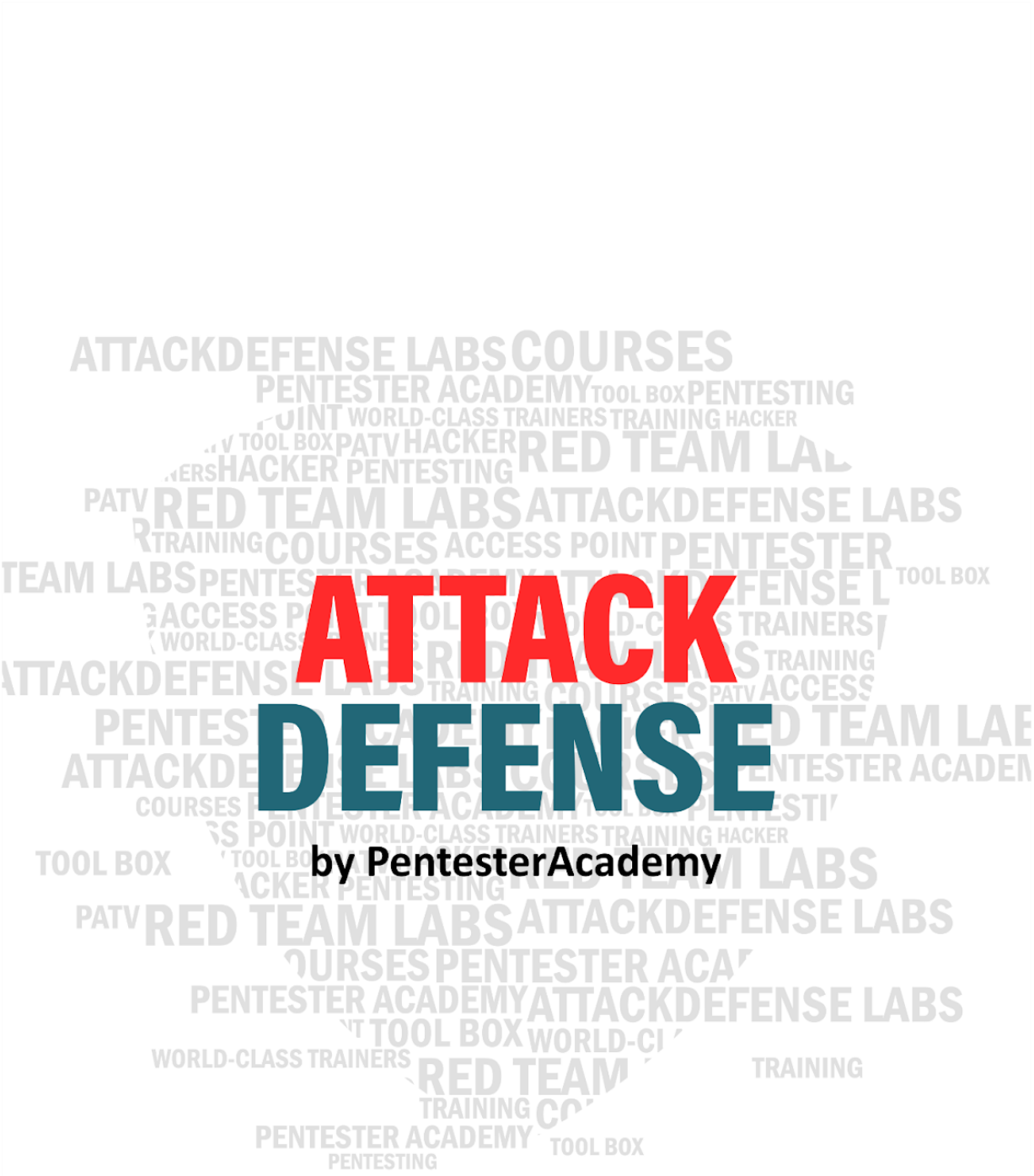
ATTACK
DEFENSE
by PentesterAcademy

Name	NodeJS Recon: Basics
URL	https://www.attackdefense.com/challengedetails?cid=686
Type	Network Recon : Webservers

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.

Q1. Find the port used by NodeJS web server and debugger.

Answer:
Web Server: 80
Debugger: 9229

Command: nmap -sC -p- 192.166.161.3

```
root@attackdefense:~# nmap -sC -p- 192.166.161.3
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-26 11:19 UTC
Nmap scan report for l006x2ijrjp2geeciohswws93.temp-network_a-166-161 (192.166.161.3)
Host is up (0.000012s latency).
Not shown: 65533 closed ports
PORT     STATE SERVICE
80/tcp   open  http
|_http-title: Site doesn't have a title (text/html).
9229/tcp open  unknown
MAC Address: 02:42:C0:A6:A1:03 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.94 seconds
root@attackdefense:~#
```

Identifying web server by sending HTTP GET request to port 80 on target machine

Command: curl 192.166.161.3

```
root@attackdefense:~# curl 192.166.161.3
Welcome to NodeJS basics lab
root@attackdefense:~#
```

Command: curl 192.166.161.3:9229

```
root@attackdefense:~# curl 192.166.161.3:9229
WebSockets request was expected
root@attackdefense:~#
```

Q2. What content is returned when a query is made to the base directory of the target server?

Answer: Welcome to NodeJS basics lab

Command: curl 192.166.161.3

```
root@attackdefense:~# curl 192.166.161.3
Welcome to NodeJS basics lab
root@attackdefense:~#
```

Q3. The server will reveal a flag when request is made for a specific directory path (e.g. http://<target_ip>/hello). There are two such flags. Retrieve both flags. The directories are present in this directory list: /usr/share/metasploit-framework/data/wordlists/directory.txt

Answer:
dc9b89e07c198dee9d576d0fcaa0cf8b
cbfbbcae6d07410ccceac87d1a1b5355

Solution:

Option 1: Using dirb

Command: dirb http://192.166.161.3 /usr/share/metasploit-framework/data/wordlists/directory.txt

Note: Please remember to remove the preceding '/' from each of directory name entry (in the directory.txt). Without this change, dirb won't work.

Option 2: Custom wrapper script around curl

Custom Script:

```
#!/bin/bash
while read dir_name; do
  echo "Trying directory: $dir_name"
  curl http://$1$dir_name
done <$2
```

```
root@attackdefense:~# cat find.sh
#!/bin/bash
while read dir_name; do
  echo "Trying directory: $dir_name"
  curl http://$1$dir_name
done <$2

root@attackdefense:~#
```

Command: ./find.sh 192.166.161.3 /usr/share/metasploit-framework/data/wordlists/directory.txt

```
Trying directory: /Office
FLAG: dc9b89e07c198dee9d576d0fcaa0cf8b
Trying directory: /Site
Welcome to NodeJS basics lab
Trying directory: /Admin
FLAG: cbfbbcae6d07410ccceac87d1a1b5355
Trying directory: /etc
Welcome to NodeJS basics lab
root@attackdefense:~#
```

Q4. The server will reveal the flag when a request is made from a specific browser to "/secret" directory. Retrieve the flag.

Answer: 6b90a6c44bc153177cb6e4ddf727b587

Command: curl -H "User-Agent: Mozilla Firefox" 192.166.161.3/secret

```
root@attackdefense:~# curl -H "User-Agent: Mozilla Firefox" 192.166.161.3/secret
FLAG: 6b90a6c44bc153177cb6e4ddf727b587
root@attackdefense:~#
```

Q5. Find the name of main app file used by the NodeJS web server.

Answer: /app/node.js

Command: curl http://192.166.161.3:9229/json/list

```
root@attackdefense:~# curl http://192.166.161.3:9229/json/list
[ {
  "description": "node.js instance",
  "devtoolsFrontendUrl": "chrome-devtools://devtools/bundled/js_app.html?experiments=true&v8only=true&ws=192.166.161.3:9229/269ab5ac-dcc5-432e-b570-d4135bbbd6b7",
  "devtoolsFrontendUrlCompat": "chrome-devtools://devtools/bundled/inspector.html?experiments=true&v8only=true&ws=192.166.161.3:9229/269ab5ac-dcc5-432e-b570-d4135bbbd6b7",
  "faviconUrl": "https://nodejs.org/static/favicon.ico",
  "id": "269ab5ac-dcc5-432e-b570-d4135bbbd6b7",
  "title": "node.js",
  "type": "node",
  "url": "file:///app/node.js",
  "webSocketDebuggerUrl": "ws://192.166.161.3:9229/269ab5ac-dcc5-432e-b570-d4135bbbd6b7"
} ]

root@attackdefense:~#
```

Q6. List the number of scripts loaded (including node-internals) on the NodeJS web server using node-inspect.

Answer: 88

Command: node-inspect 192.166.161.3:922

```
root@attackdefense:~# node-inspect 192.166.161.3:9229
debug> scripts(true)
  9: internal/bootstrap/loaders.js
  10: internal/bootstrap/node.js
  11: events.js <native>
  12: internal/async_hooks.js <native>
  13: internal/errors.js <native>
  14: util.js <native>
  15: internal/util/inspect.js
  16: internal/util.js <native>
  17: internal/util/types.js <native>
  18: internal/validators.js
  19: internal/encoding.js <native>
  20: buffer.js <native>
  21: internal/buffer.js <native>
  22: internal/process/per_thread.js
  23: internal/process/main_thread_only.js
  24: internal/process/stdio.js <native>
```

```
  83: internal/dgram.js
  84: internal/socket_list.js <native>
  85: internal/cluster/round_robin_handle.js <native>
  86: internal/cluster/utils.js <native>
  87: internal/cluster/shared_handle.js <native>
  88: internal/cluster/worker.js <native>
```

Q7. A breakpoint is set in the main app file. Trigger the breakpoint and use node-inspect debugger to view the source code of the main app file. The source contains a "Hidden Flag". Retrieve the flag.

Answer: a656ddfc746cf3cd1bb2c454793d8e48

Solution:

Interacting with the debugger:

Command: node-inspect 192.166.161.3:9229

```
root@attackdefense:~# node-inspect 192.166.161.3:9229
debug>
debug>
```

Triggering the breakpoint. Open the lab link in new tab/window to get access to another terminal window and run the command given below:

Command: curl 192.166.161.3

```
root@attackdefense:~# curl 192.166.161.3
```

The breakpoint will hit on the debugger session

```
root@attackdefense:~# node-inspect 192.166.161.3:9229
break in file:///app/node.js:4
  2 var useragent=require('useragent');
  3 var server = http.createServer(function (req, res) {   //create web server
> 4     debugger;
  5     //Hidden Flag: a656ddfc746cf3cd1bb2c454793d8e48
  6     if (req.url == "/Office") {
debug>
```

References:

1. NodeJS (https://nodejs.org/en/)
2. node-inspect (https://www.npmjs.com/package/node-inspect)