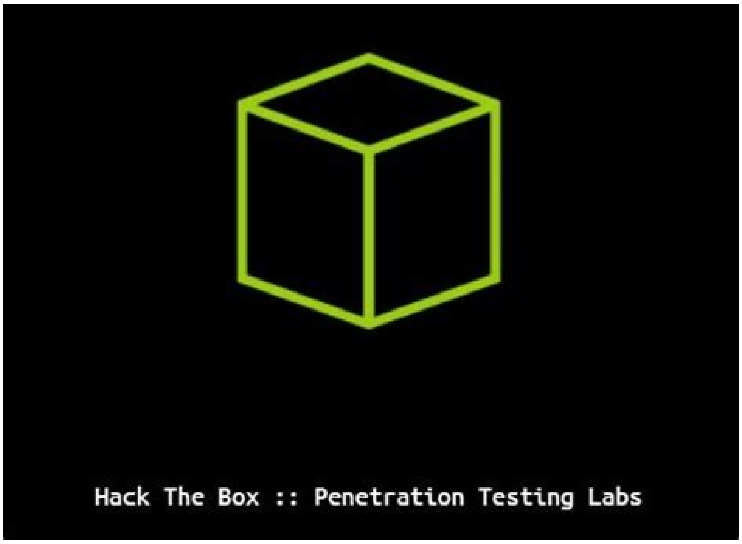
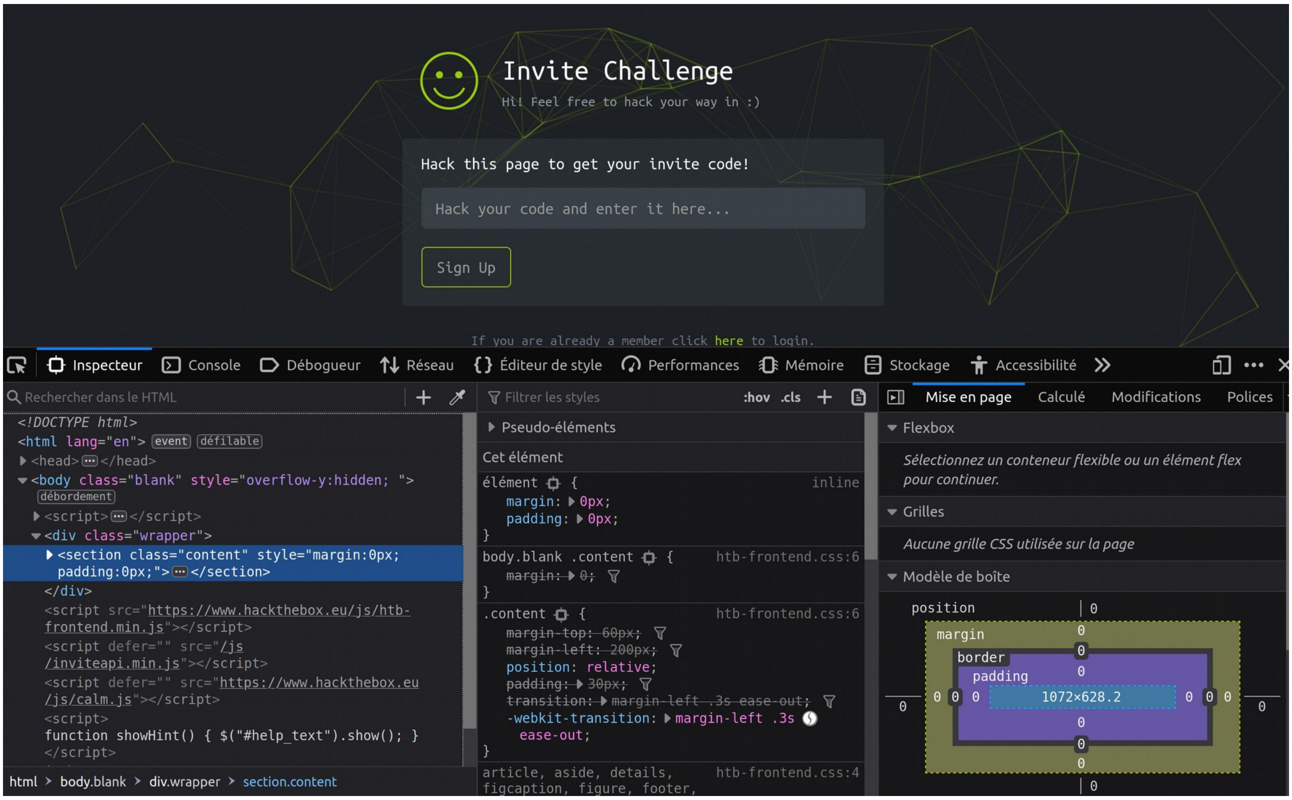
Hack the box is an online penetration testing labs to train your skill and exchanges with other information security enthousiasts.



To have an account on HTB we need to «Hack» the site to it’s will give us a code to activate an account. I will to give detail how I’ve had the code.

1:

I’ve used developpers tools of Mozilla Firefox on the home page of HTB:



Invite Challenge

Hi! Feel free to hack your way in :)

Hack this page to get your invite code!

Hack your code and enter it here...

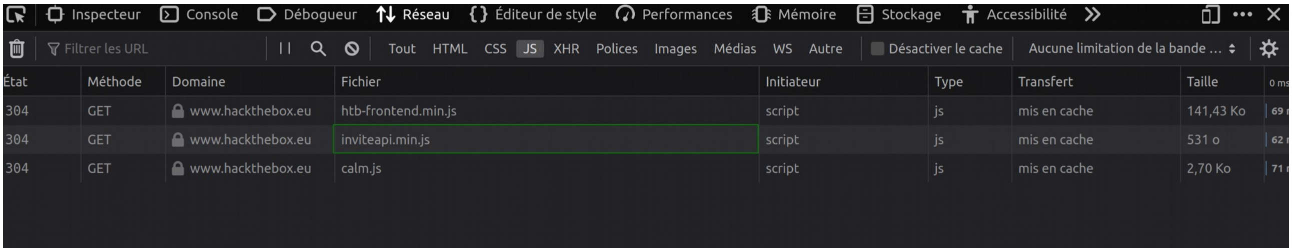
Sign Up

If you are already a member click here to login.

```
<!DOCTYPE html>
<html lang="en"> event défilable
  <head>…</head>
  <body class="blank" style="overflow-y:hidden; ">
  débordement
    <script>…</script>
    <div class="wrapper">
      <section class="content" style="margin:0px;
      padding:0px;">…</section>
    </div>
    <script src="https://www.hackthebox.eu/js/htb-
    frontend.min.js"></script>
    <script defer="" src="/js
    /inviteapi.min.js"></script>
    <script defer="" src="https://www.hackthebox.eu
    /js/calm.js"></script>
    <script>
    function showHint() { $("#help_text").show(); }
    </script>
```

```
élément {                                  inline
    margin: 0px;
    padding: 0px;
}
body.blank .content {           htb-frontend.css:6
    ~~margin: 0;~~
}
.content {                      htb-frontend.css:6
    ~~margin-top: 60px;~~
    ~~margin-left: 200px;~~
    position: relative;
    ~~padding: 30px;~~
    ~~transition: margin-left .3s ease-out;~~
    -webkit-transition: margin-left .3s
      ease-out;
}
article, aside, details,        htb-frontend.css:4
figcaption, figure, footer,
```

I used network tabs and use filter to check JS files:



État	Méthode	Domaine	Fichier	Initiateur	Type	Transfert	Taille
304	GET	www.hackthebox.eu	htb-frontend.min.js	script	js	mis en cache	141,43 Ko
304	GET	www.hackthebox.eu	inviteapi.min.js	script	js	mis en cache	531 o
304	GET	www.hackthebox.eu	calm.js	script	js	mis en cache	2,70 Ko

We can see that HTB give us a JS script named inviteapi.min.js.

2:

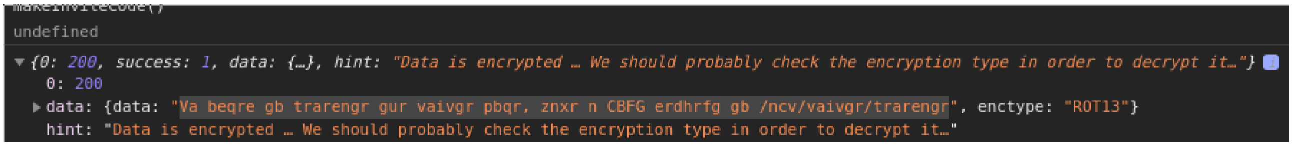
I inspected this JS script and I saw this string at the end of the script:

```
‘function|console|log|makeInviteCode|ajax|type|POST|dataType|json|
url||api|invite|how|to|generate|success|error'.split('|'),0,{}
```

The split function replace space in a string by the character gives in paramater, so | replace space.

3:

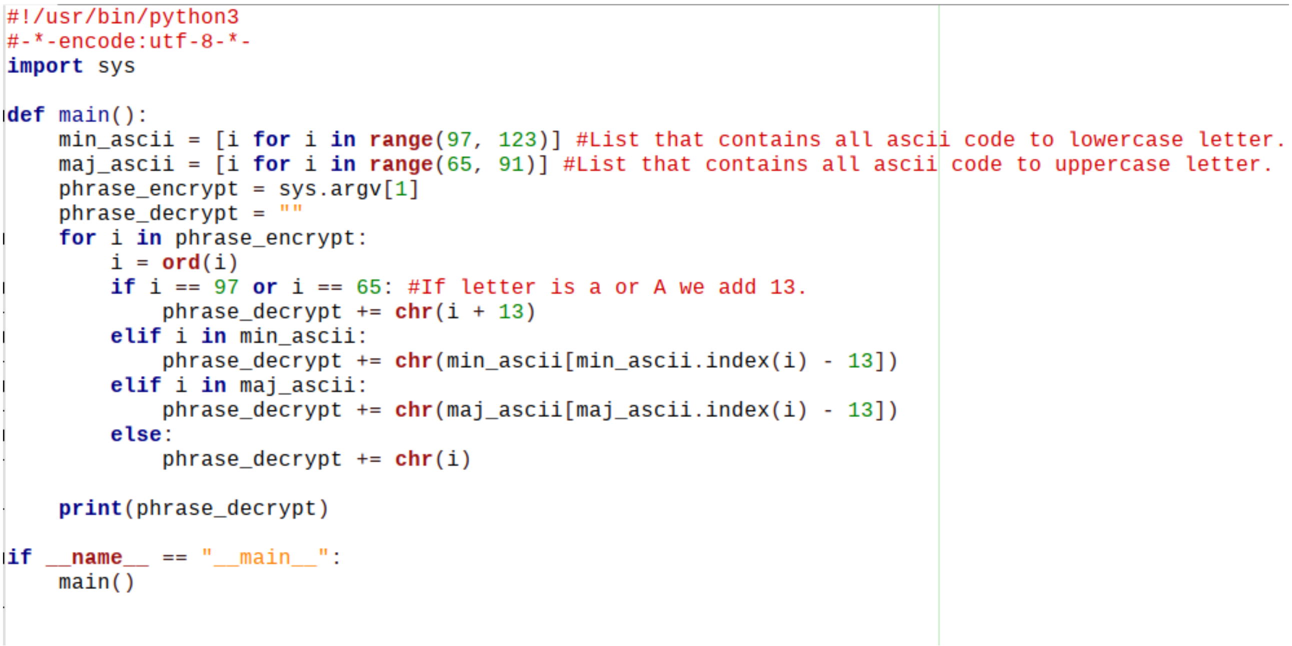
The string say us go to console and use the function makeInviteCode:



```
makeInviteCode()
undefined
▼{0: 200, success: 1, data: {…}, hint: "Data is encrypted … We should probably check the encryption type in order to decrypt it…"}
   0: 200
  ▶data: {data: "Va beqre gb trarengr gur vaivgr pbqr, znxr n CBFG erdhrfg gb /ncv/vaivgr/trarengr", enctype: "ROT13"}
   hint: "Data is encrypted … We should probably check the encryption type in order to decrypt it…"
```

The data is encrypted with ROT13 which is a kind of ceasar encryption, we rotate letter by 13 places.

To decrypt this I written a little python script:



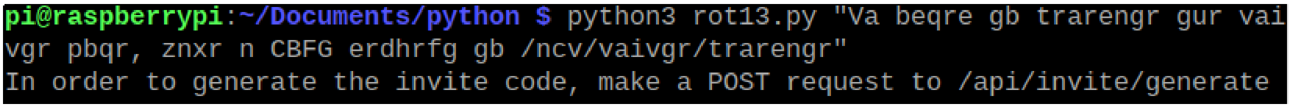
```python
#!/usr/bin/python3
#-*-encode:utf-8-*-
import sys

def main():
    min_ascii = [i for i in range(97, 123)] #List that contains all ascii code to lowercase letter.
    maj_ascii = [i for i in range(65, 91)] #List that contains all ascii code to uppercase letter.
    phrase_encrypt = sys.argv[1]
    phrase_decrypt = ""
    for i in phrase_encrypt:
        i = ord(i)
        if i == 97 or i == 65: #If letter is a or A we add 13.
            phrase_decrypt += chr(i + 13)
        elif i in min_ascii:
            phrase_decrypt += chr(min_ascii[min_ascii.index(i) - 13])
        elif i in maj_ascii:
            phrase_decrypt += chr(maj_ascii[maj_ascii.index(i) - 13])
        else:
            phrase_decrypt += chr(i)

    print(phrase_decrypt)

if __name__ == "__main__":
    main()
```

Data decrypted is:

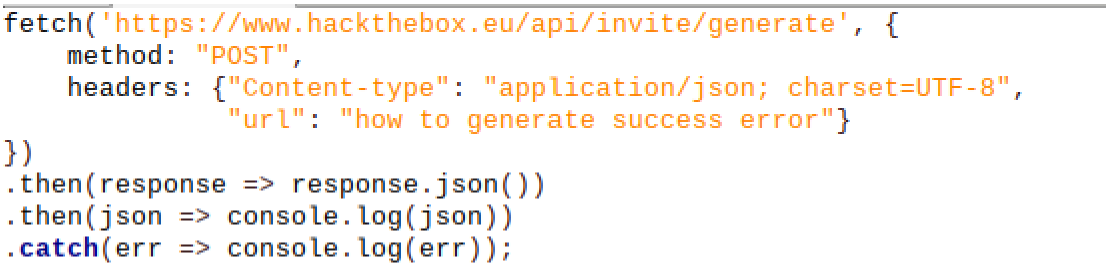


```
pi@raspberrypi:~/Documents/python $ python3 rot13.py "Va beqre gb trarengr gur vaivgr pbqr, znxr n CBFG erdhrfg gb /ncv/vaivgr/trarengr"
In order to generate the invite code, make a POST request to /api/invite/generate
```

4:

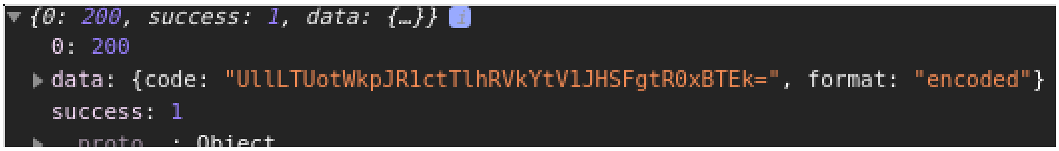
So, the message say us to sent POST request to the server, for this I will use Fetch API’s Javascript, which is easy to use version of XMLHttpRequest.

Here is my POST request:



```
fetch('https://www.hackthebox.eu/api/invite/generate', {
    method: "POST",
    headers: {"Content-type": "application/json; charset=UTF-8",
              "url": "how to generate success error"}
})
.then(response => response.json())
.then(json => console.log(json))
.catch(err => console.log(err));
```

Here is the response:



```
▼ {0: 200, success: 1, data: {…}} i
   0: 200
 ▶ data: {code: "UllLTUotWkpJR1ctTlhRVkYtV1JHSFgtR0xBTEk=", format: "encoded"}
   success: 1
 ▶ __proto__: Object
```

5:

We can saw that the code is encoded in base64 due the equal at the end of the encoded data.
To decode the encoded data I used the decoder of Base64 Guru.
So the code is...