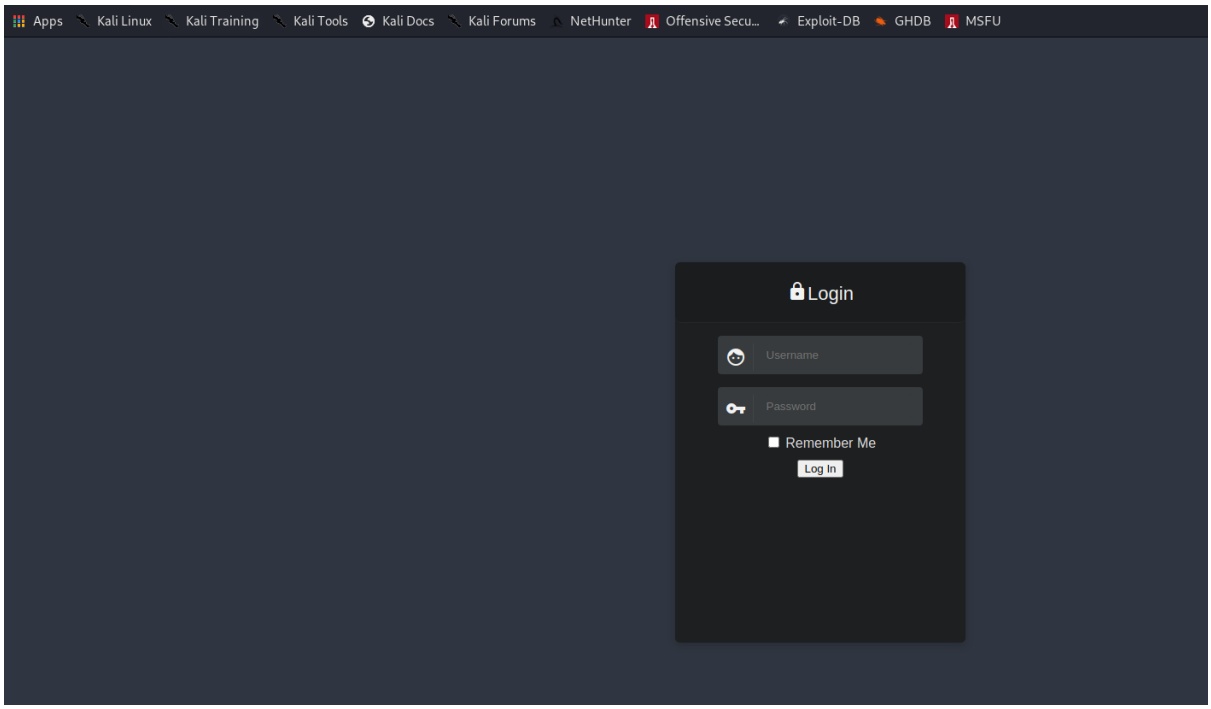
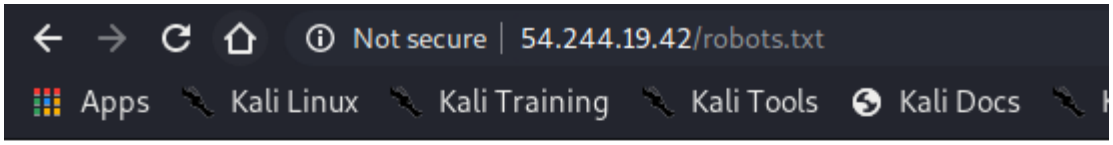
Walkthrough by Sneha Surana

A login page with user name and password we need them to get enter into account



Since we don't have both username and password let check robots.txt that tells about allowed and disallowed path




```
User-agent: *
Disallow: /dev/
          /dev/login.php
```

We got to disallowed But lets check it

This page only accepts POST request

Both the paths were not useful.Let’s have look on Source code

```
<script>
    function loginFunction() {
        var username = document.getElementById("username").value;
        var password = document.getElementById("password").value;
        var x = password.slice(0,9);
        var y = password.slice(9);
        var z = md5(y);
        console.log("reached");
        if (x == "\x43\x6C\x6F\x75\x64\x53\x45\x4B\x5F")
        {
            if (z == "06a3cccaafedc5b09b10b4b26f02a9e1")
            {
                //document.getElementById("msg").innerHTML = "Right";
                window.location = "./loader.php?p=bWVzc2FnZTFfdG9famFyZWQudHh0Cg%3D%3D&password=" + password;
            }
            else
            {
                document.getElementById("msg").innerHTML = "Try harder!";
            }
        }
        else
        {
            document.getElementById("msg").innerHTML = "Incorrect credentials";
        }
    }
</script>
</html>
```

After understanding the code password is been sliced from 0 to 9 and stored in variable x and again password is sliced and stored to variable y and now md5 of the same

We got md5 z==0683…… and X=’\x43\...

Let’s decode it

Original code:

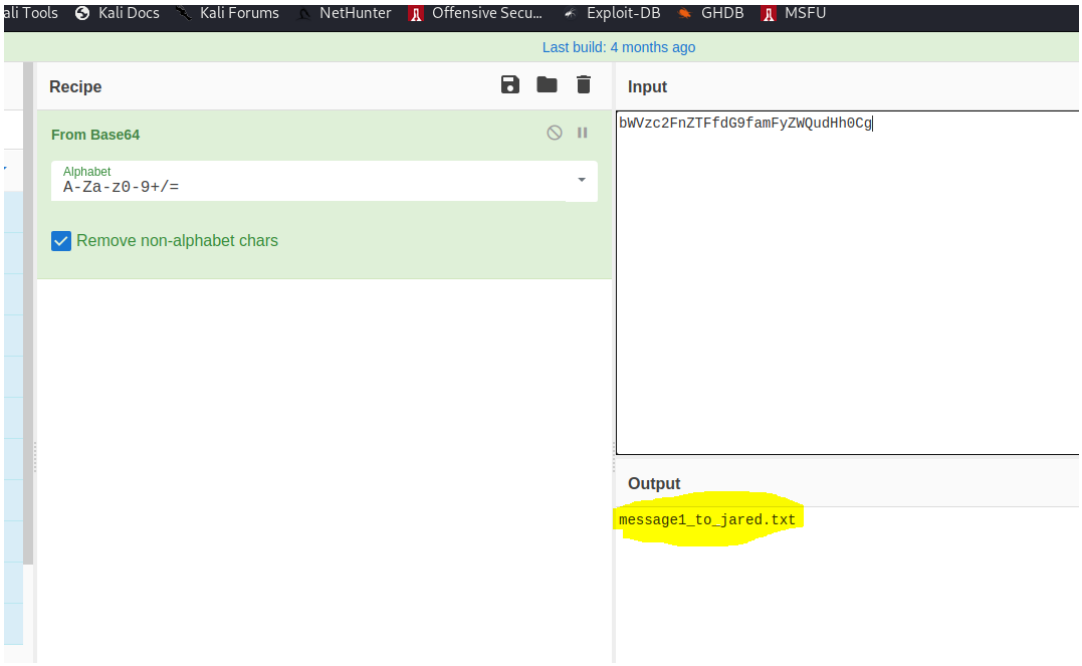
Decoded results:

Since password is being sliced so that final password is **CloudSEK_jeniffer**

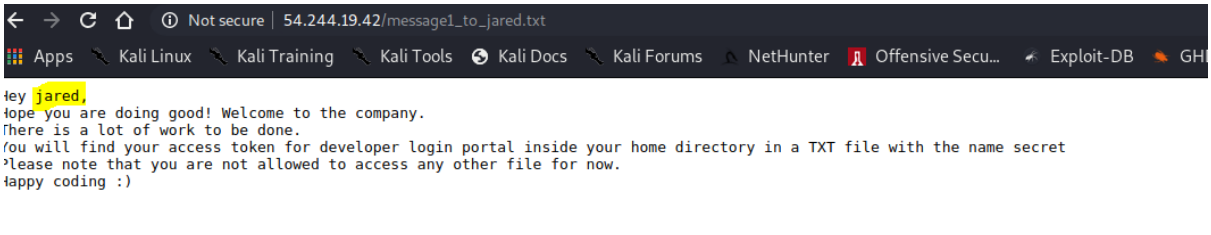
But we don’t haver user let look again at source code

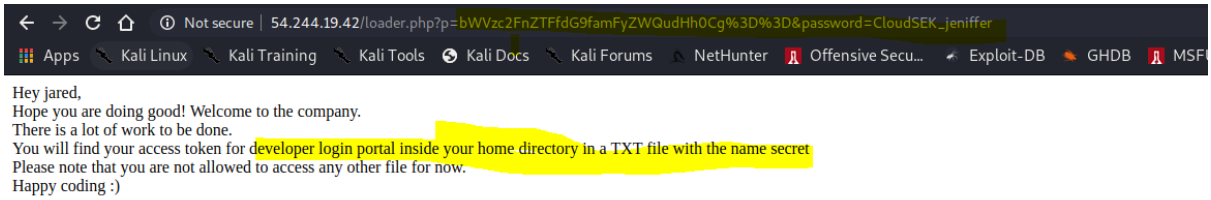
```
if (x == "\x43\x6C\x6F\x75\x64\x53\x45\x4B\x5F")
{
    if (z == "06a3cccaafedc5b09b10b4b26f02a9e1")
    {
        //document.getElementById("msg").innerHTML = "Right";
        window.location = "./loader.php?p=bWVzc2FnZTFfdG9famFyZWQudHh0Cg%3D%3D&password=" + password;
    }
    else
    {
```

We got base64 code Lets' decode it

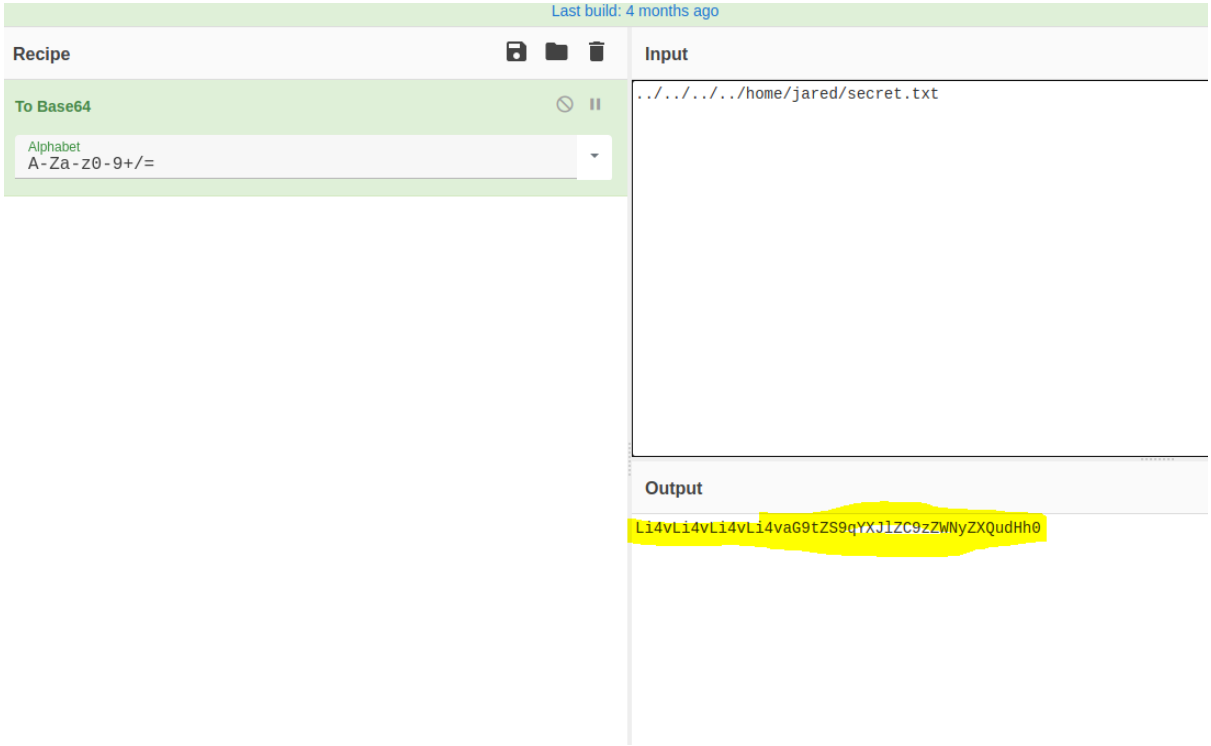


After decoding we got a file and got a message

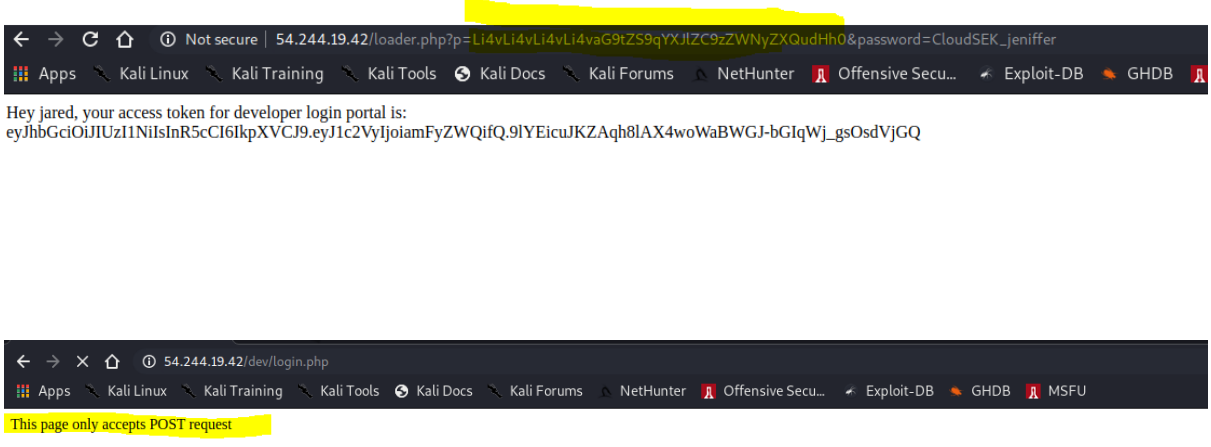




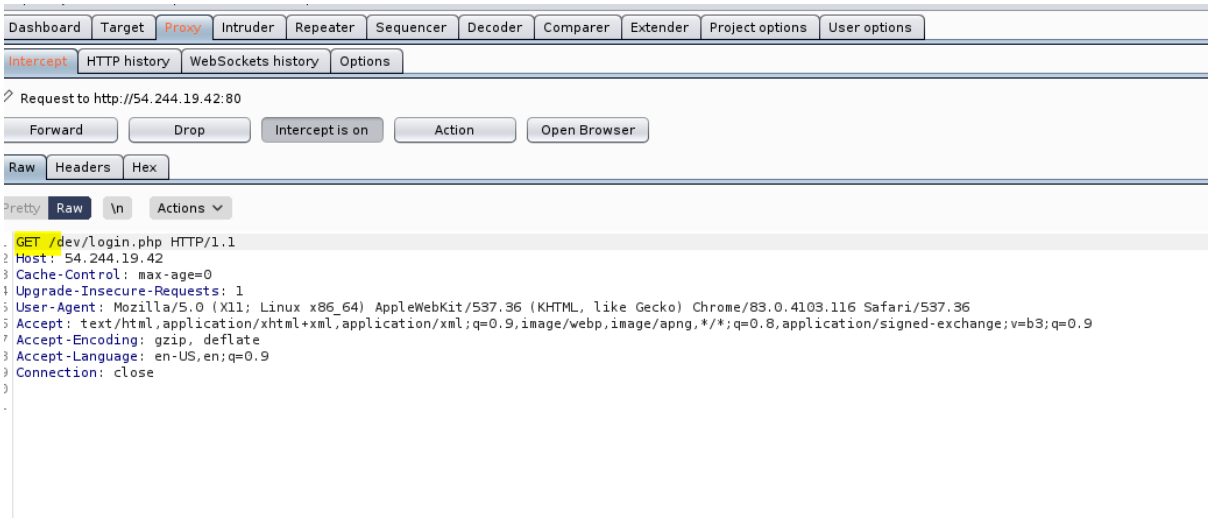
Since we need to fine secret file in home directory Let use LDAP injection and try to add '../../../../home/jared/secret.txt' . But it doesn't worked and in the source code it accepting base64 Let convert '../../../../home/jared/secret.txt to base 64



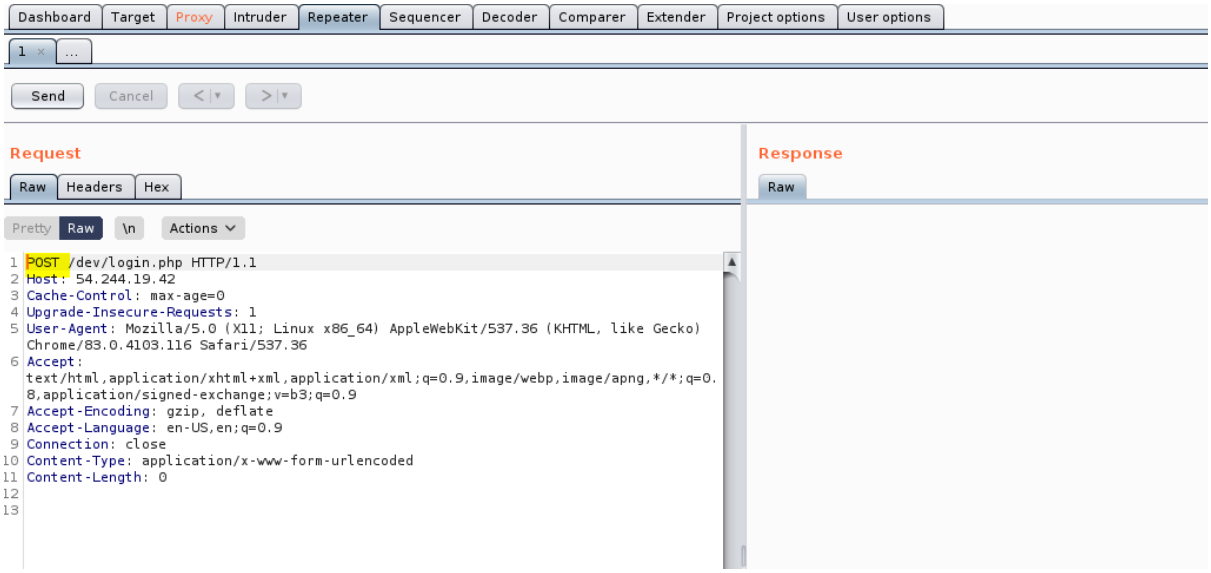
And put in p= .After that we got access token as in hint it was mention about JWT



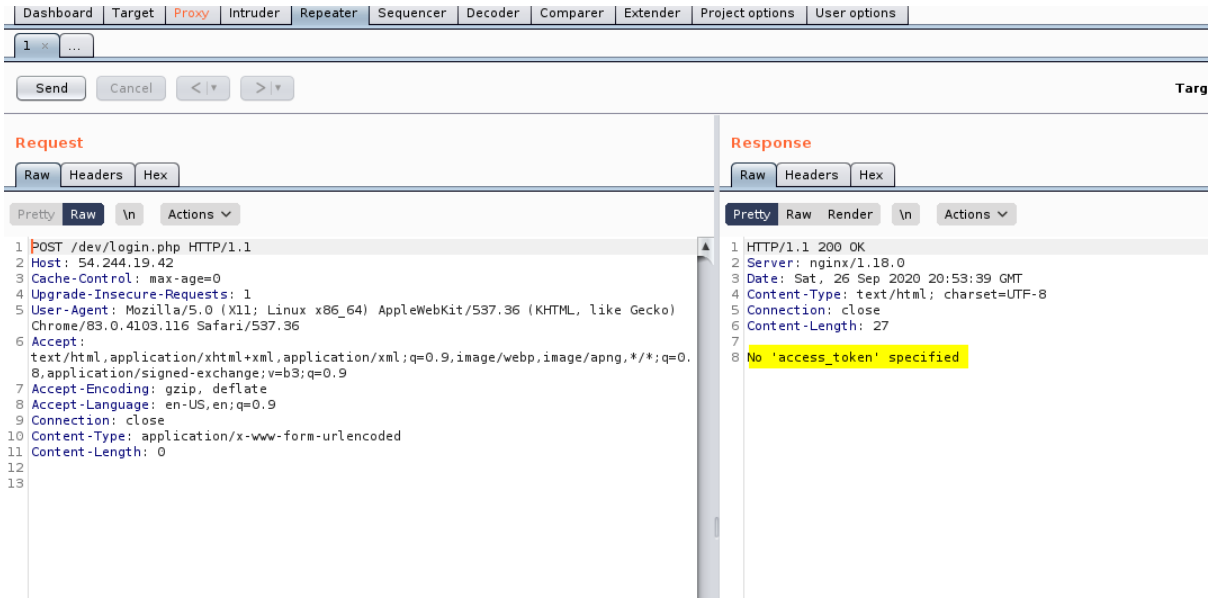
Let get the request normally in burpsuite by enabling foxy -proxy and send the request to repeater



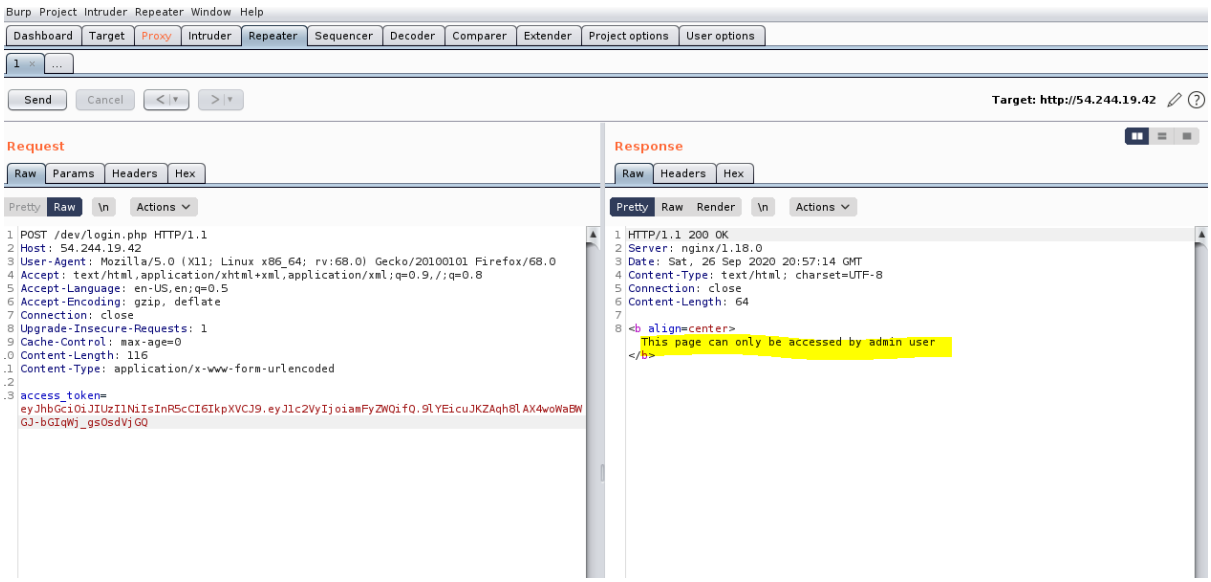
Click On Action button and Send to Repeater to make post request and change the request from GET to POST



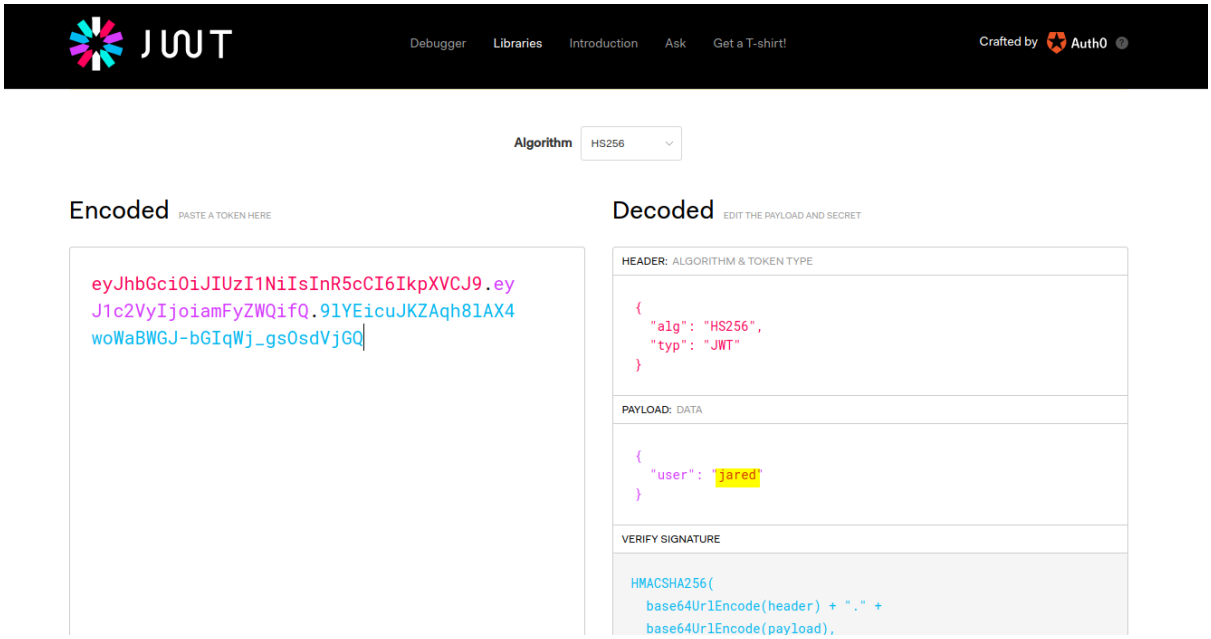
Click on Send button



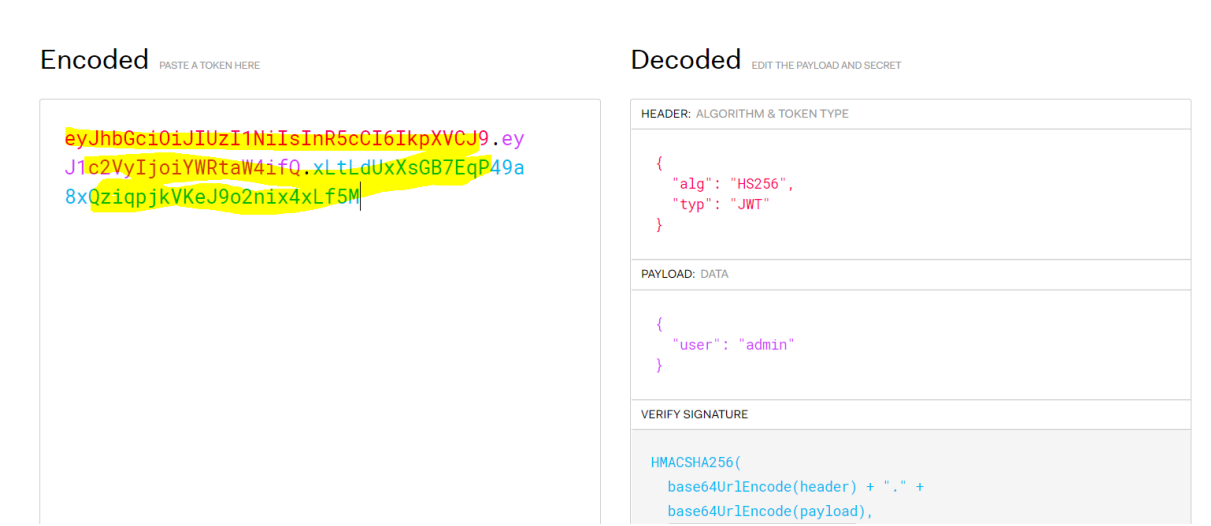
We need to add access_token in the Request .Now specify the access token in the request and post it We need to post request by admin user

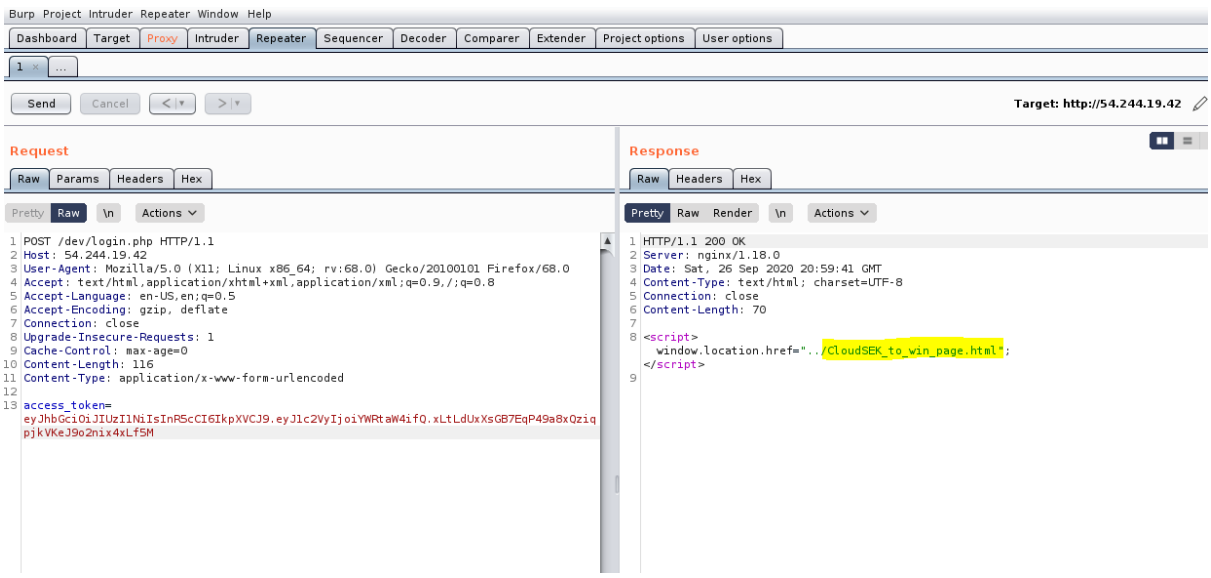


Now as in hints JWT was mentioned let analyse access_token in it

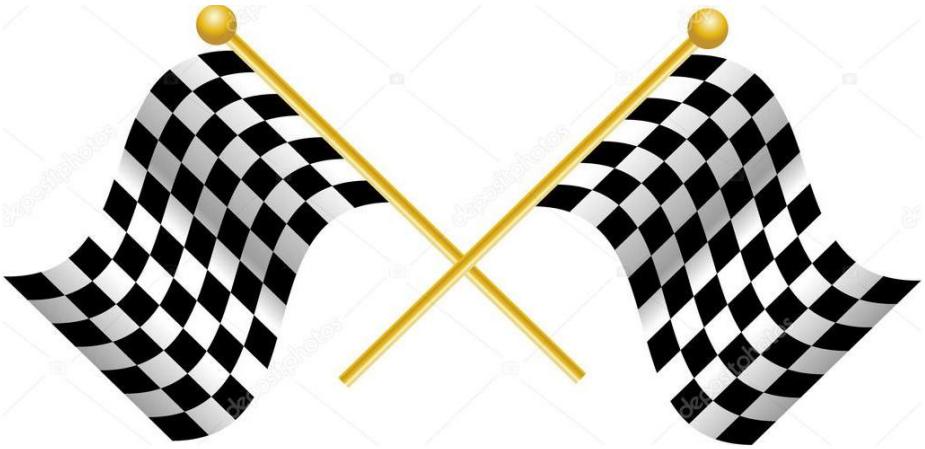


Now change user to admin and new access_token is generated Now let post request using it





We got page and now we have analyse as mentioned in hint

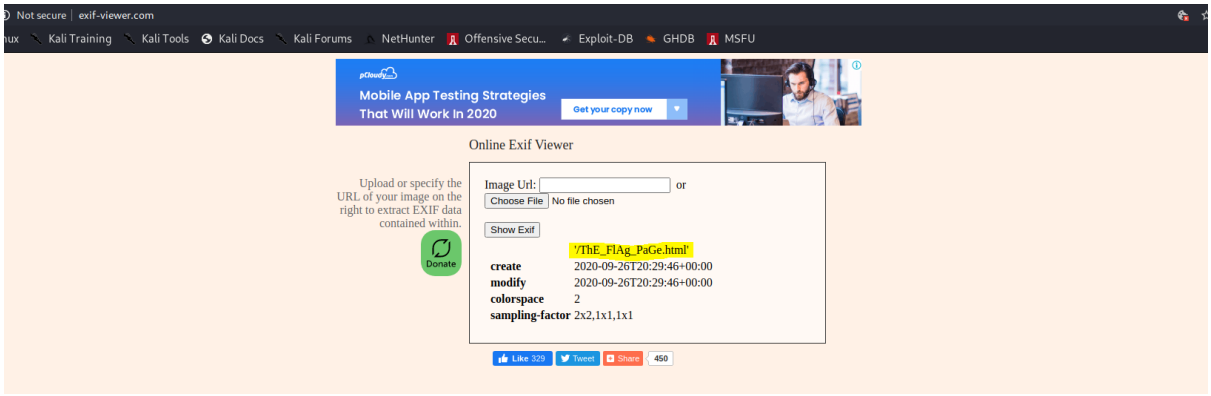


Ha ha ha! You are close! But the game isn't over yet!

If you will look in the dark, you will find your worth!

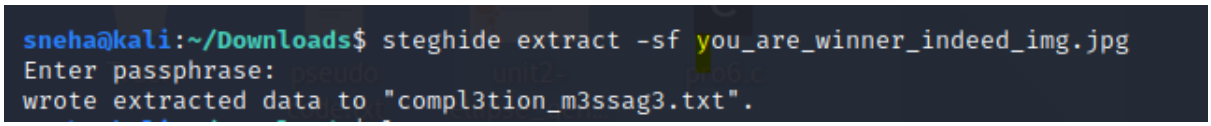
You can be a winner!

Let analyse this image and After using steghide ,zsteg, foremost and binwalk no result and examin meta data used online tool



We got again a new page and let enter

Again image this time there is flag also Run zsteg foremost binwalk no result now let run steghide and we have flag this can be password let's try



Extracted the data got a file