Web Application Security Testing -> **Report for the 20231218 lesson 3**

**Content**

 Web Application Security Testing -> **Report for the 20231218 lesson 3 ** **Authorization bypass**

 **Insecure Direct Object Reference**

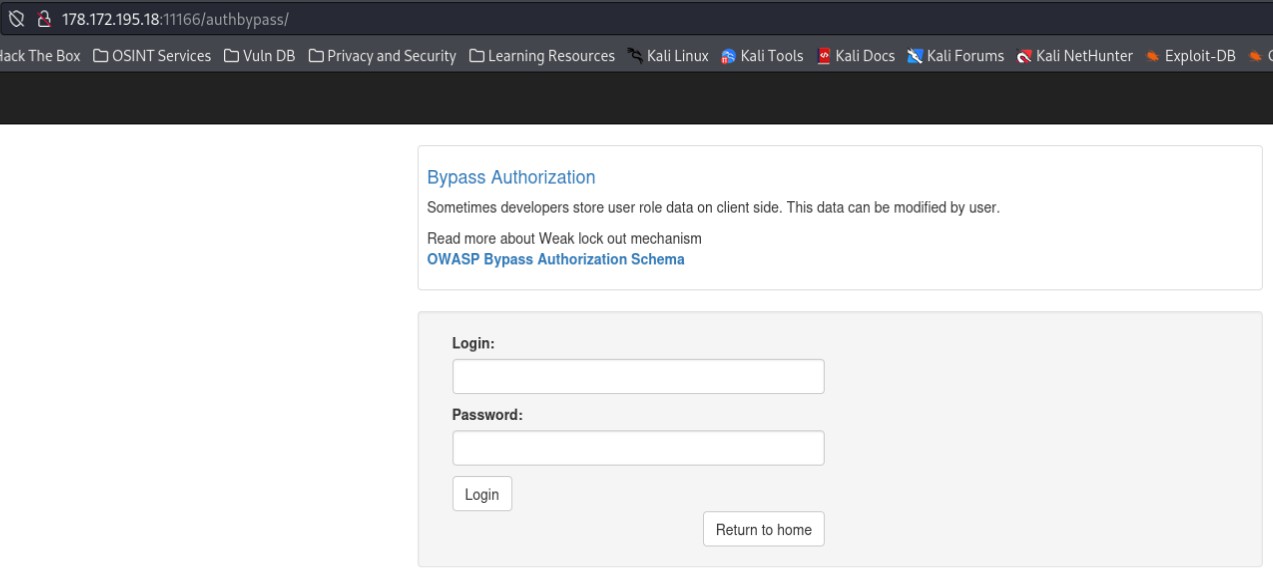
 **Broken Access Control** <- *I missed the access time.*

 **Weak Lockout** <- *I missed the access time.*

# Authorization bypass

Solution

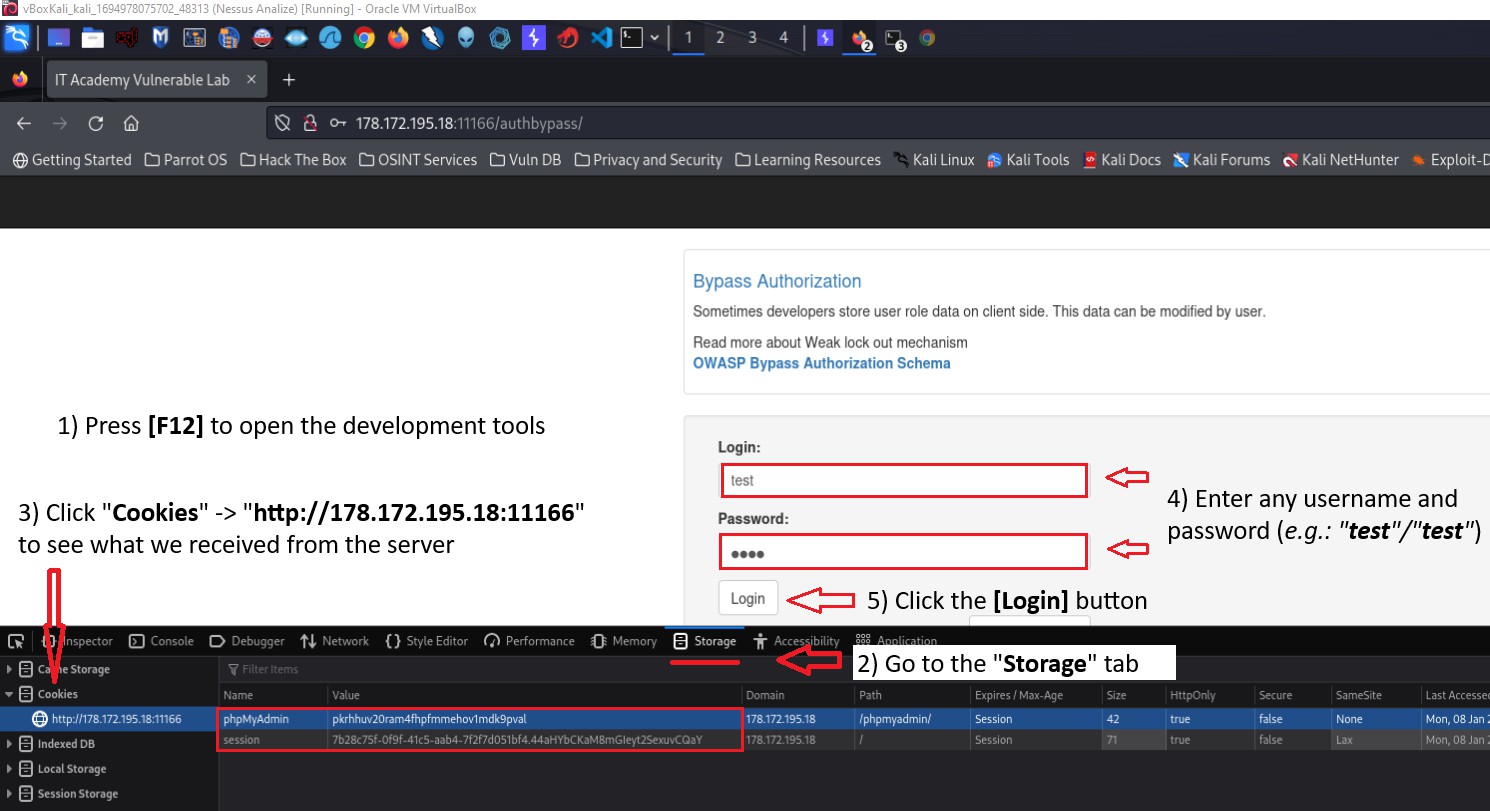
1. Run the **Authorization Bypass** task.



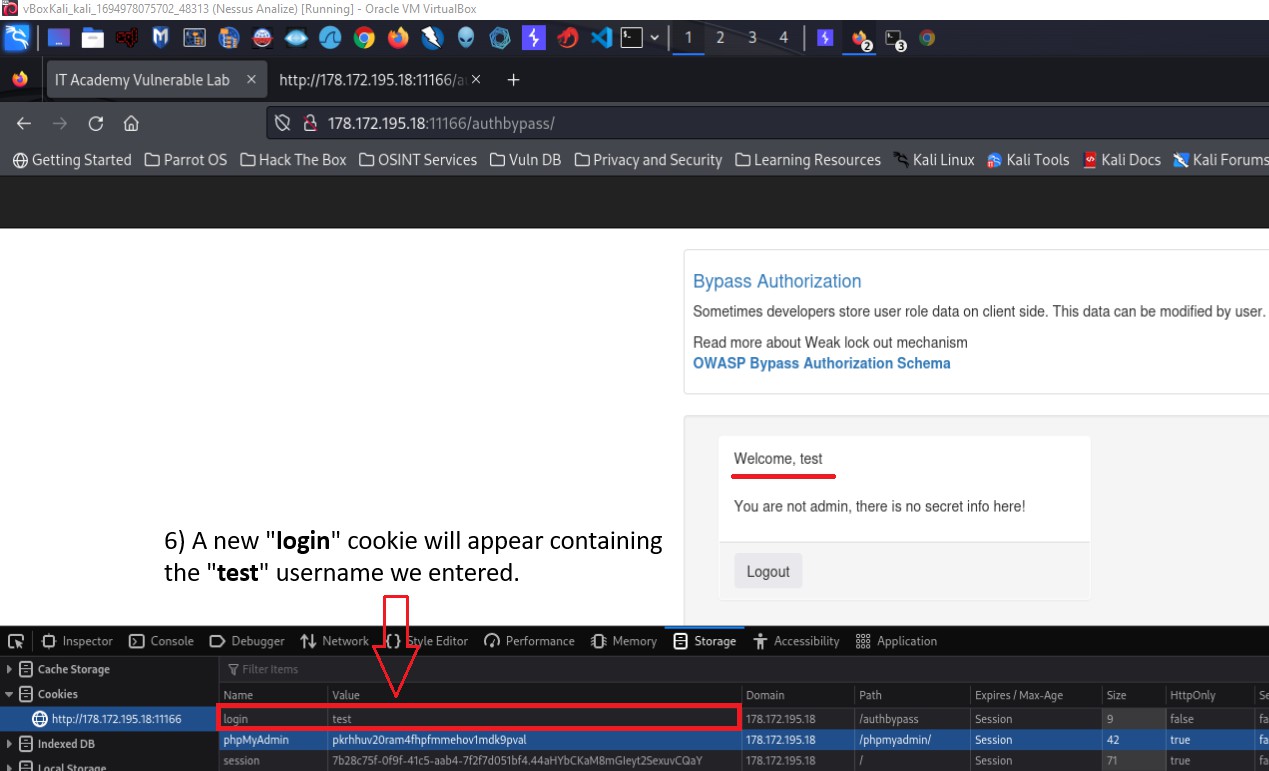
1. Press the [F12] button to open the developer tools.
2. Go to the "*Storage*" tab.
3. Click "*Cookies*" -> "*http://178.172.195.18:11166*" to see what we received from the server.

**NOTE:** We have 2 cookies: "*phpMyAdmin*" and "*session*".

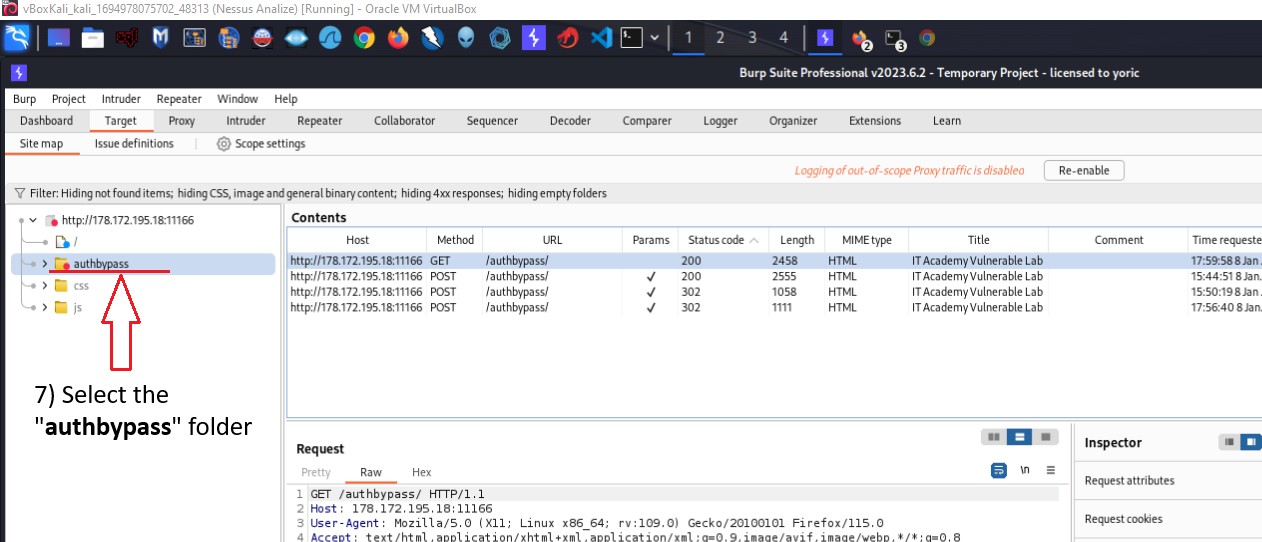
1. Enter any username and password (*e.g.: test/test*)
2. Click the [Login] button.



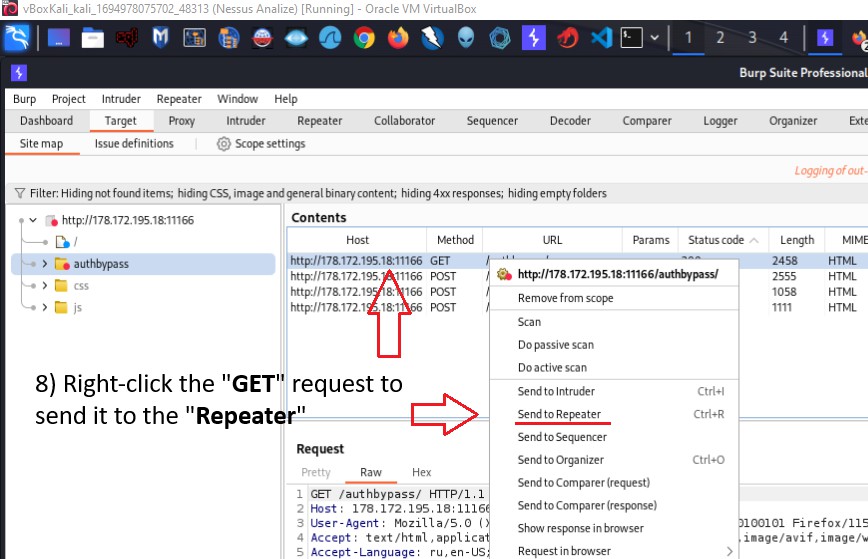
1. Once we log in, the *welcome* page appears, and a new "*login*" cookie contains the "*username*" our entered.



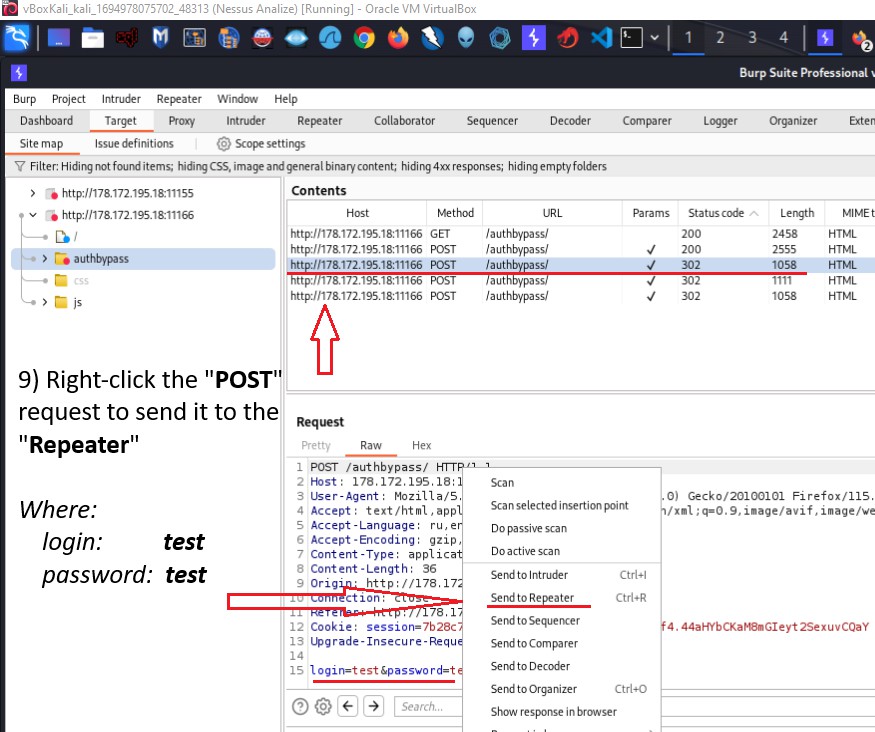
1. Run **Burpsuite** to drill down into events and compare behavior.



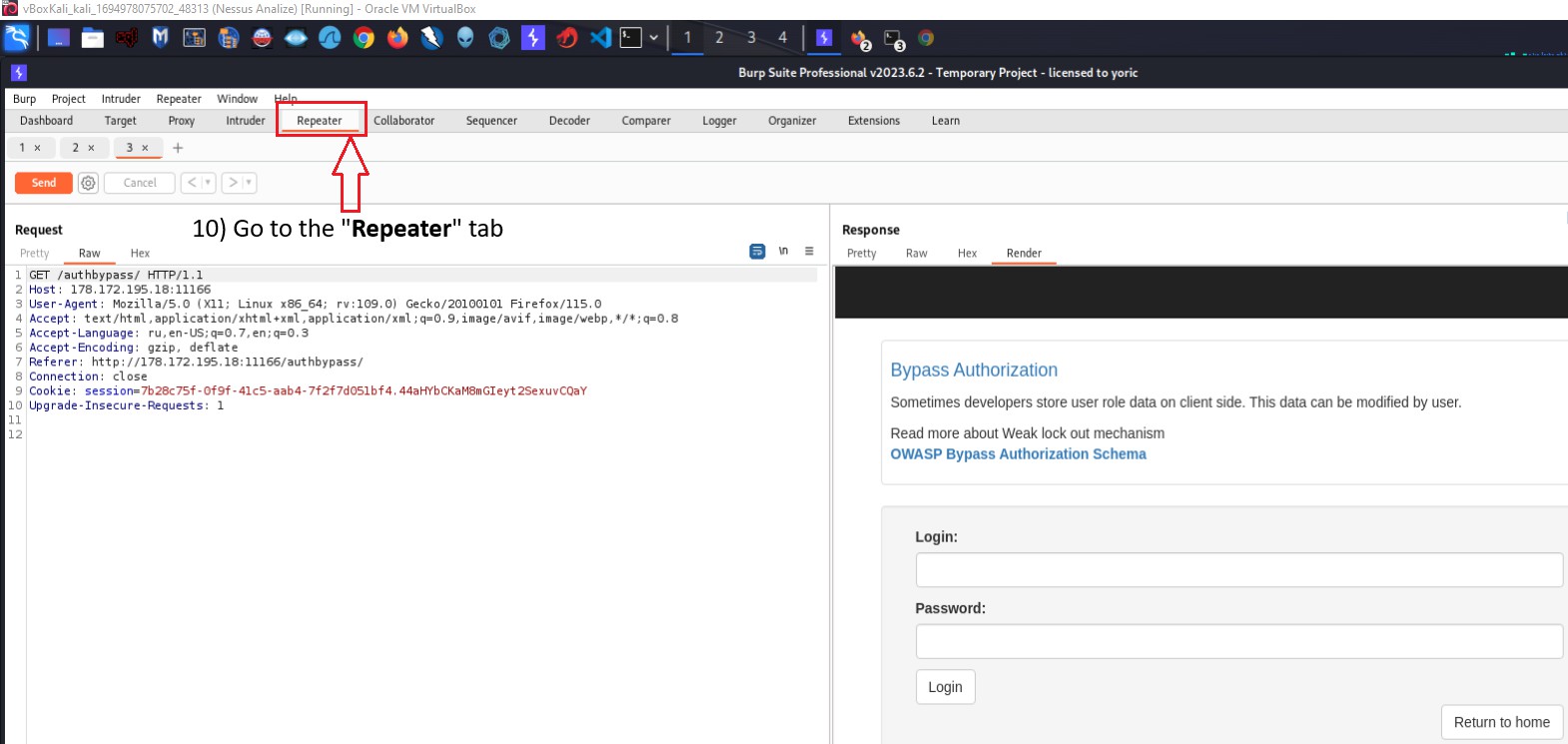
1. Expand the host tab "http://178.172.195.18:11166".
2. Select the "authbypass" folder to see the requests.
3. In the Content area, right-click the "*GET*" request to send it to the "*Repeater*".



1. Then we do the same for the "*POST*" request, to which we send the login and password (*test/test*).



1. Go to the "*Repeater*" tab.

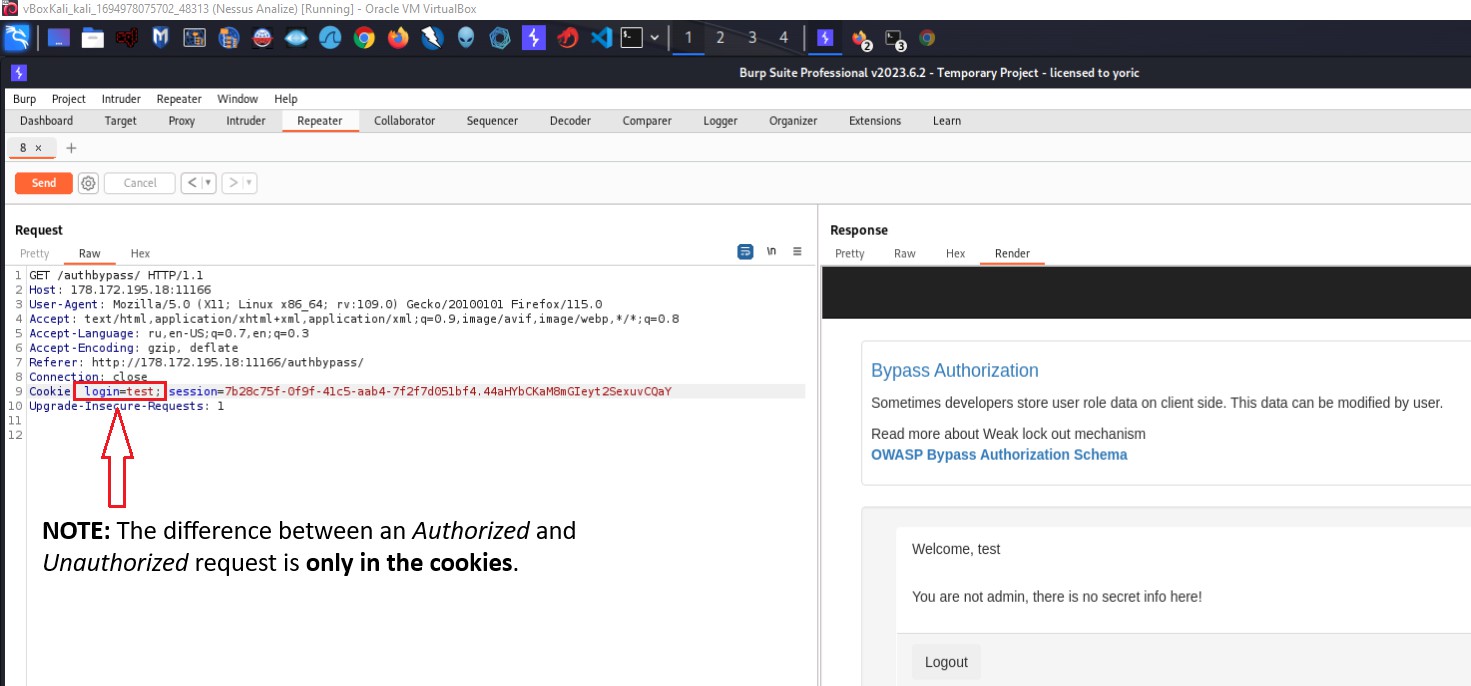


**NOTE:**

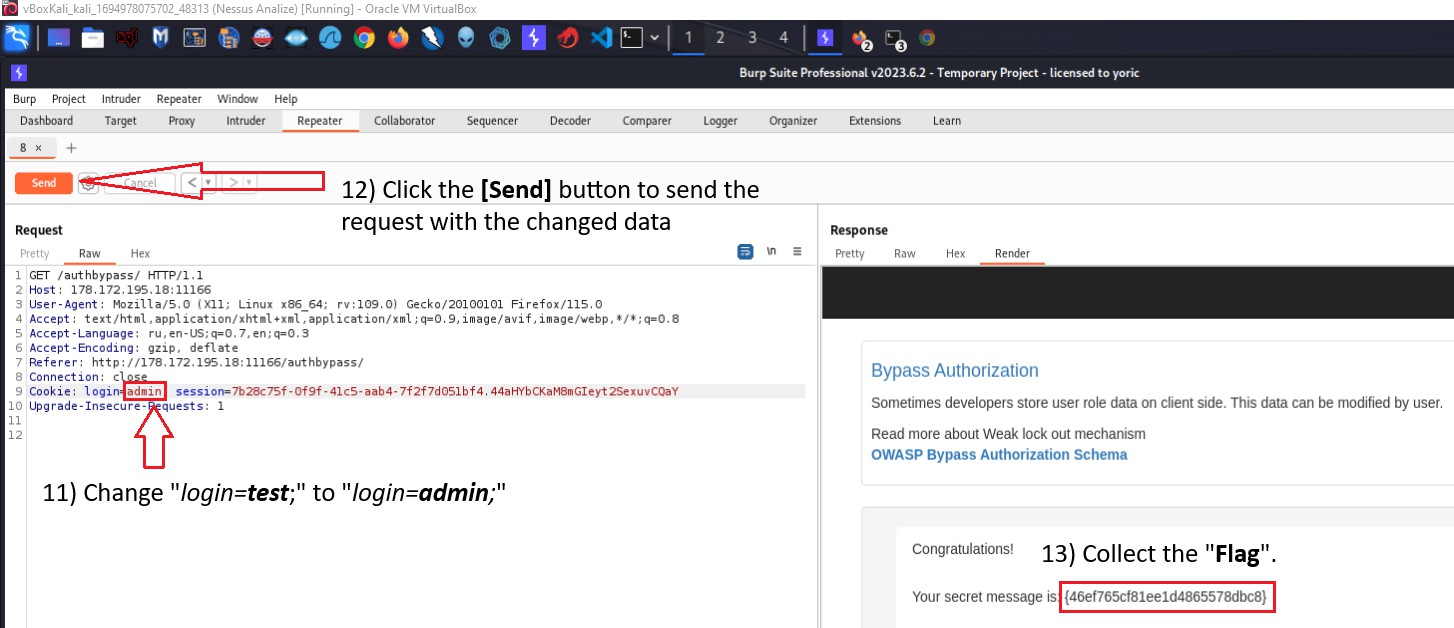
If we compare an "*Authorized*" request with an "*Unauthorized*", we will only find the difference in the cookies.

Upon "*Autorized*" request from the server, we receive a welcome page with the message "**You are not an admin, there is no secret info here!**" and the request contains cookies with a new line "login=test ;".

1. Change "login=test;" to "login=admin;";

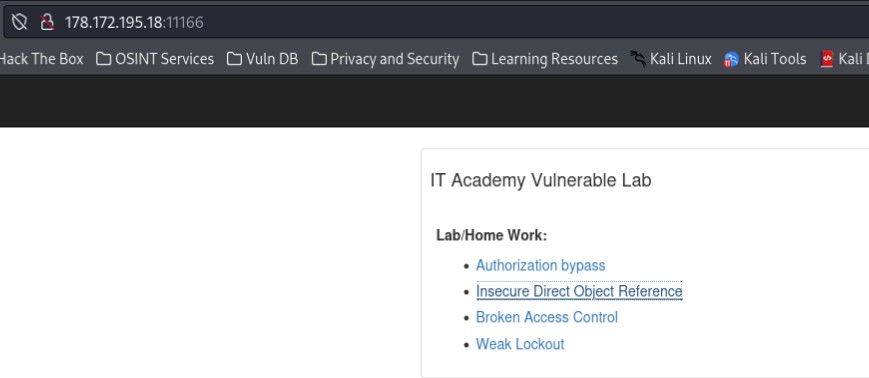


1. Click the [Send] button to resend the request with the changed data.



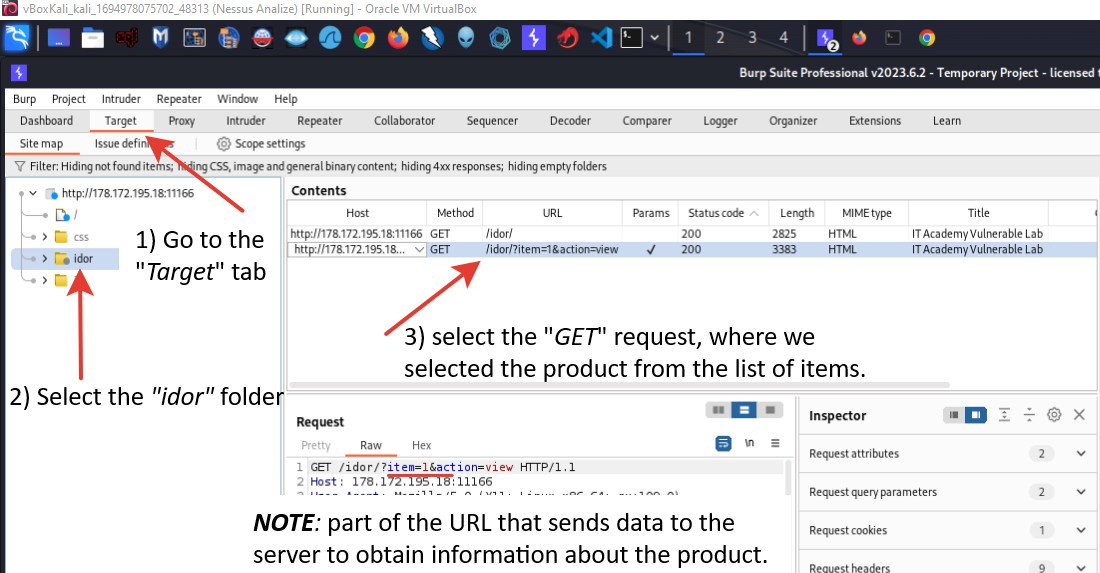
1. The answer has been changed and we see that the new welcome page contains a flag.

# Insecure Direct Object Reference



Solution

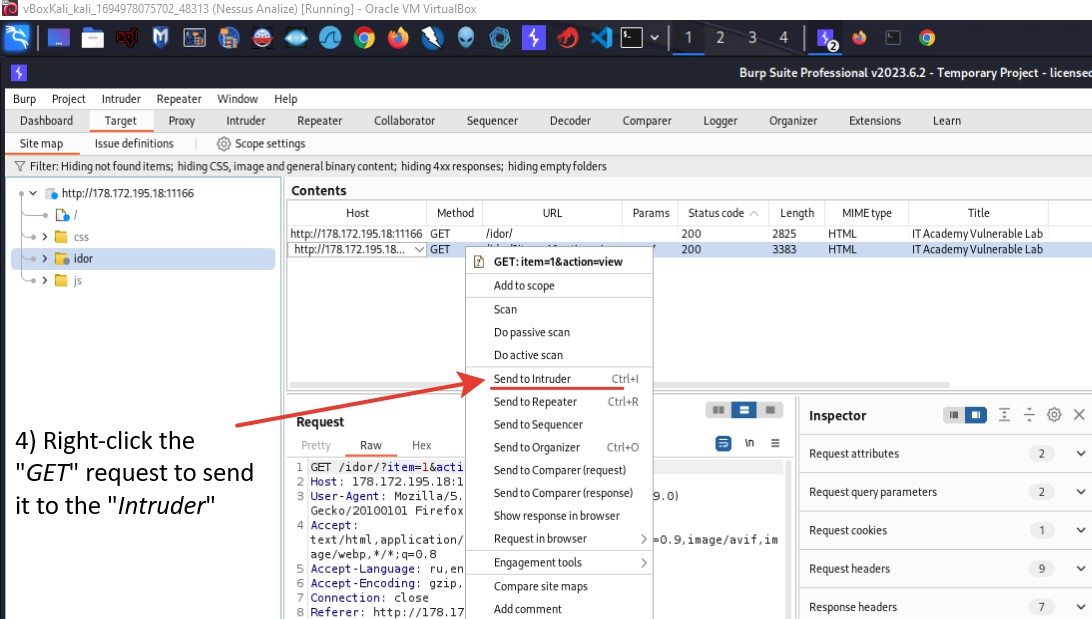
1. Launch **Burpsuite** to be able to automatically brute values.
2. Go to the "*Target*" tab.
3. Expand the host tab "http://178.172.195.18:11166".
4. Select the "*idor*" folder to see the requests.
5. In the Content area, select the "*GET*" request, where we selected the product from the list of items.



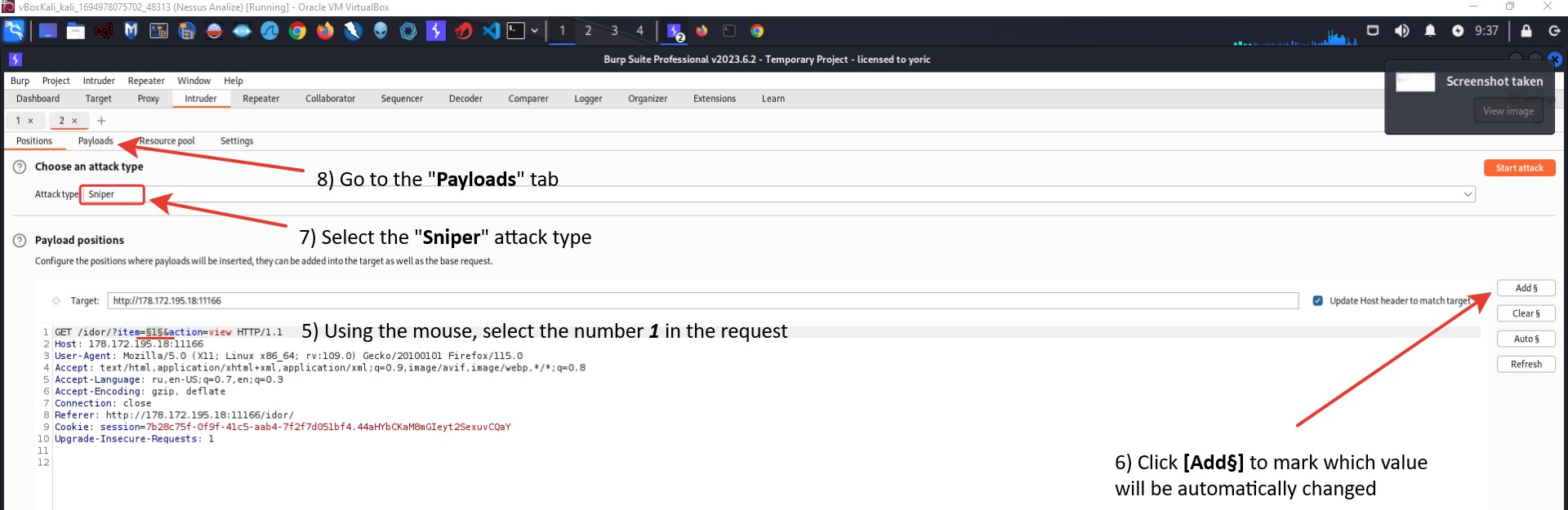
**NOTE:**

?item=1&action=view - part of the URL that sends data to the server to obtain information about the product.

1. Right-click the "*GET*" request to send it to the "*Intruder*".



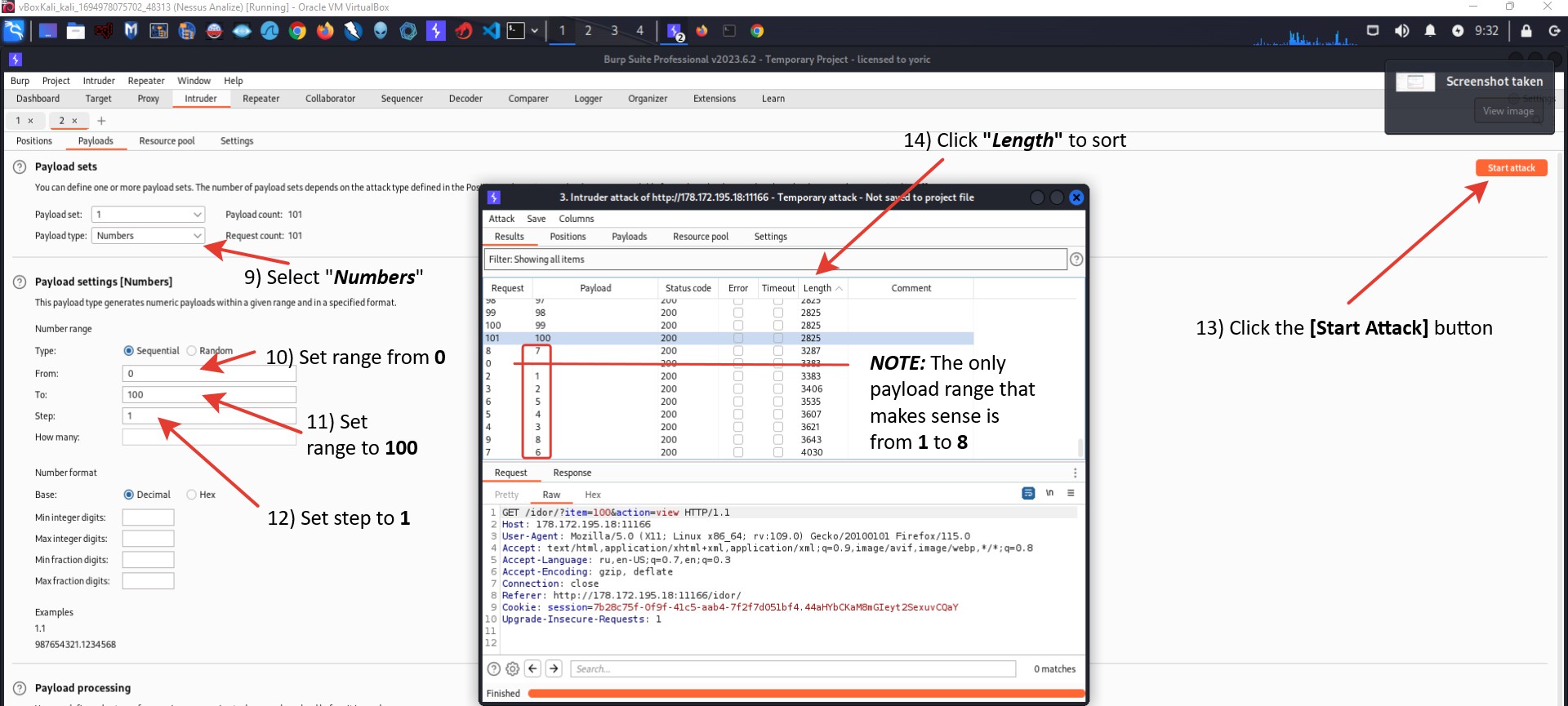
1. Go to the "*Intruder*" tab.
2. Using the mouse, select the number '1' (*depends on your choice in the product list*) in the request
3. Click [Add§] to mark which value will be automatically changed.
4. Choose an attack type "*Sniper*"



**NOTE:**

Since we will only be affecting one value, select the Sniper attack type.

1. Go to the Payloads tab.



1. Select **Payload Type** as “Numbers”.
2. Set the following properties for the "*Number Range*"  Type => Sequential

 From => 0

 To => 100

 Step => 1

1. Click the [Start Attack] button.
2. When the attack is complete, click "*Length*" to sort the length of the response and understand what content the response has.

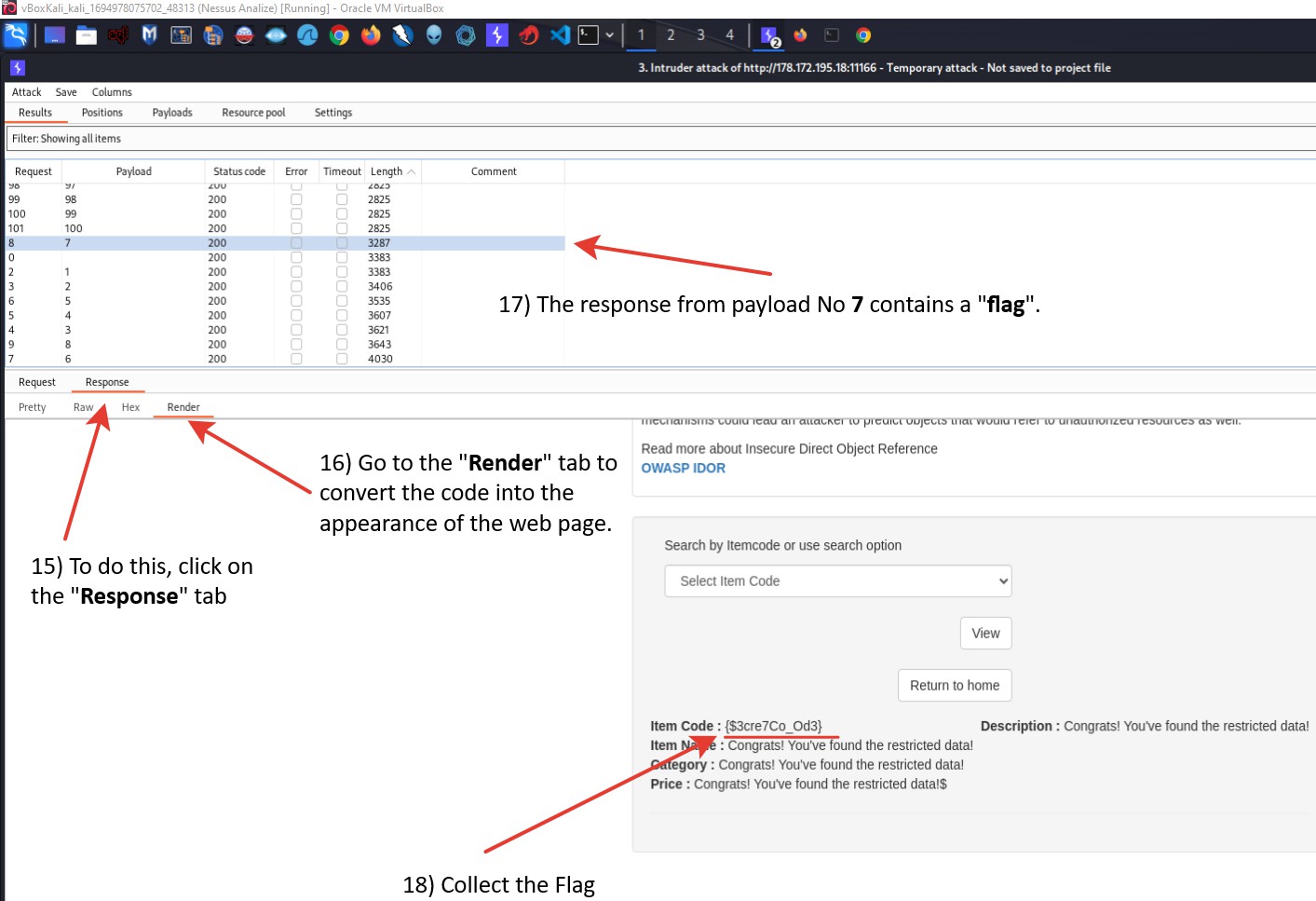
**NOTE:**

All responses of the same length tell us that there is no suitable content to view and we can ignore them. The same situation with the first and second request (*item=0 and item=1*).

Now we can see that the only payload range that makes sense is from **1** to **8**. As we remember, there are only *5* items in the item list, but after the attack **we can find 3 more**.

Let's see what they contain.

1. To do this, click on the "*Response*" tab.
2. Go to the "*Render*" tab to convert the code into the appearance of the web page.
3. Then select the payload number from 6 to 8 to view them.
4. The response from payload 7 contains a flag.



**Broken Access Control** <- *I missed the access time*

**NOTE:** I missed the time to have an access and solve this task.

**Weak Lockout** <- *I missed the access time*

**NOTE:** I missed the time to have an access and solve this task.