

3. Insecure Direct Object Reference(IDOR)

Objective: To identify and exploit IDOR vulnerabilities

Tools: Burpsuit, DVWA, custom web application

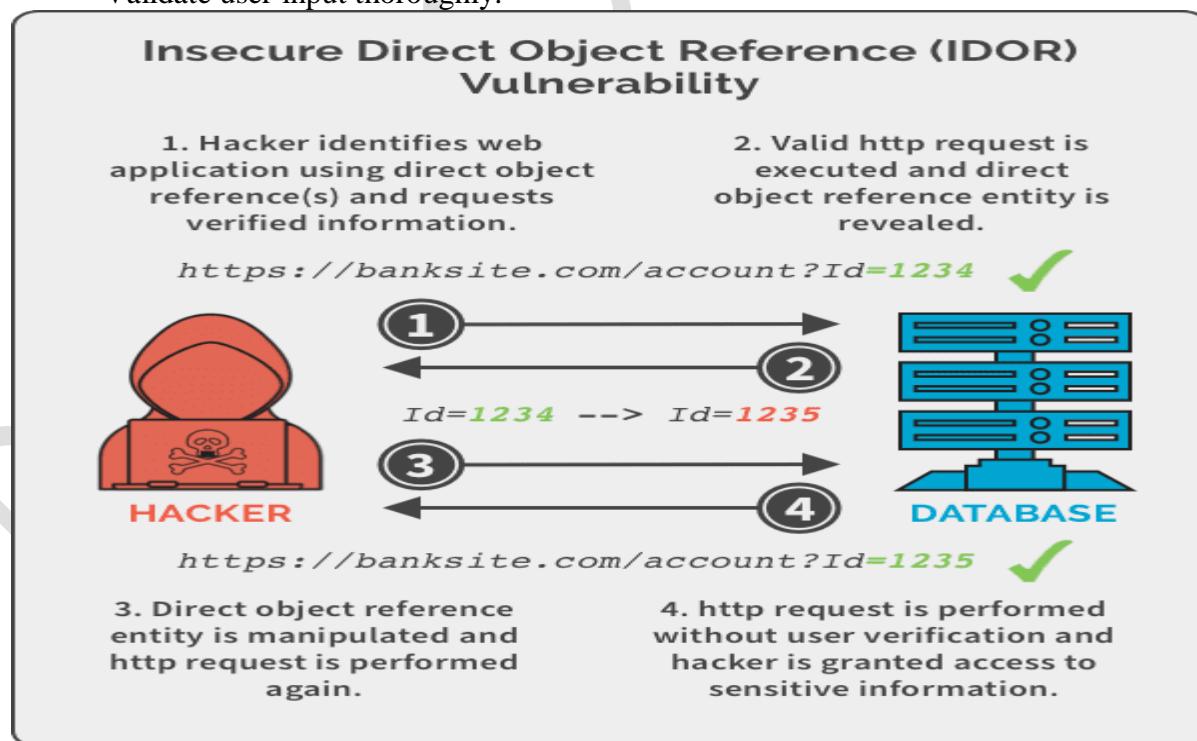
Insecure Direct Object Reference (IDOR) is a web application security vulnerability that occurs when an application exposes internal object identifiers to users without proper access controls. This allows attackers to manipulate these identifiers and gain unauthorized access to sensitive data.

Here are some examples of IDOR:

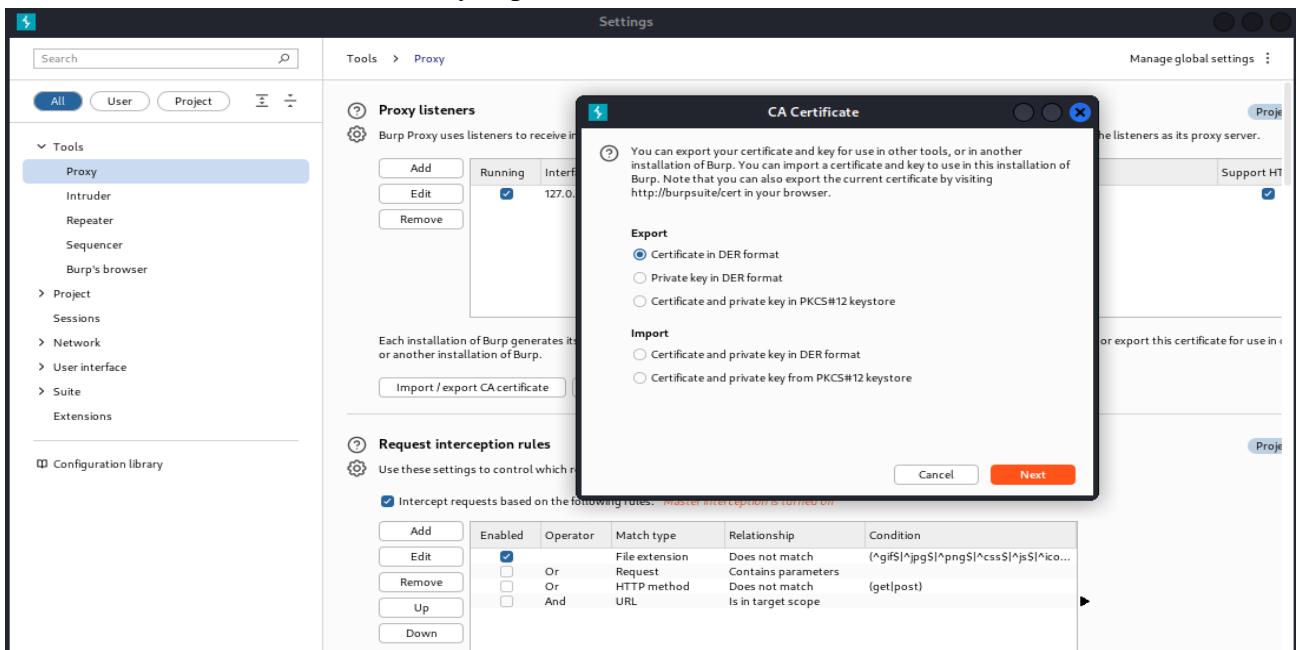
- Direct reference to database objects: An attacker can modify a customer number value to bypass access controls and view other customers' records.
- Substituting a parameter value: A malicious hacker can substitute a parameter value in a URL with a similar value that belongs to another user.

To prevent IDOR, you can:

- Use a UUID instead of a direct user ID in a URL.
- Use Role-Based Access Control (RBAC) to define specific roles and permissions for different users.
- Enforce strict authentication requirements like multi-factor authentication (MFA) or strong password policies.
- Validate user input thoroughly.

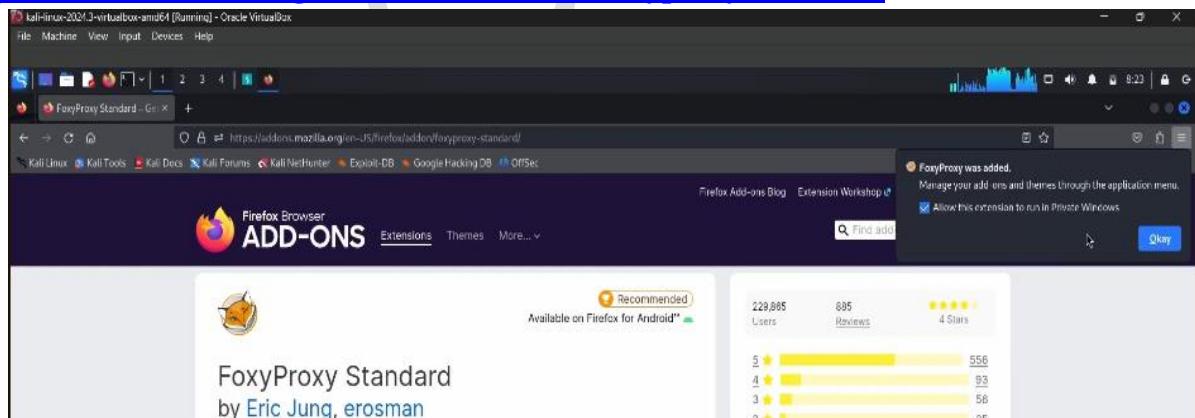


Step1: Open the burp suit in kali, navigate to setting in burp and click on to import/export button. In export select first option and click on next and select the file, place it on desktop with cert file name. Now certificate is successfully exported



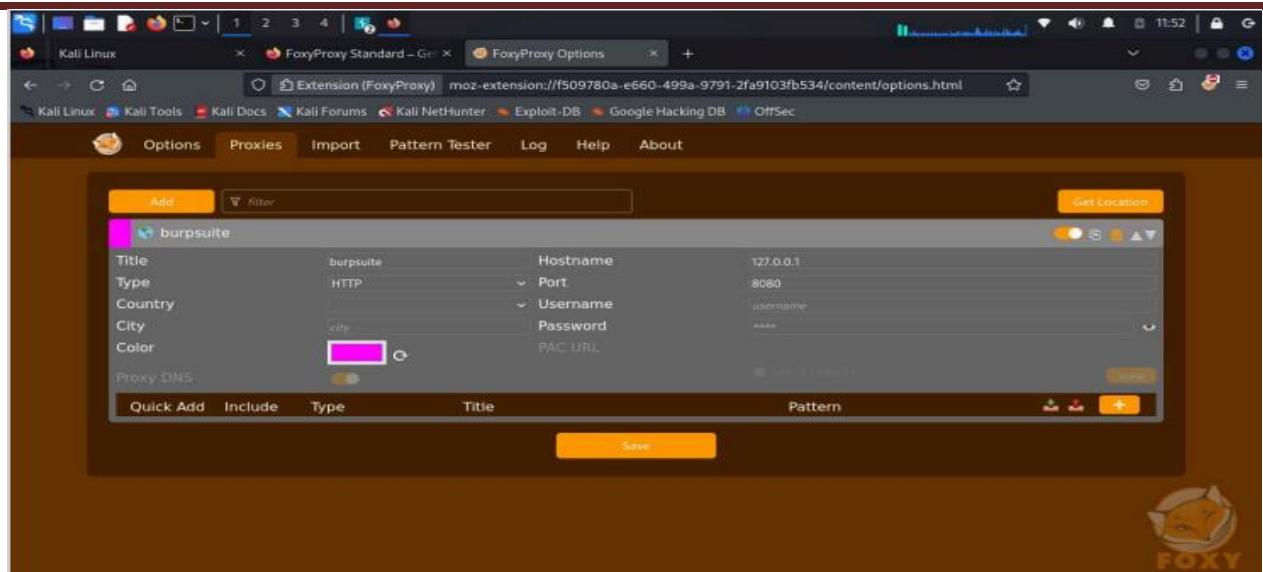
Step2: Go to the Firefox browser in kali, search for the extension **foxyproxy** in google and click on add Firefox button. Then go to tool bar and add foxyproxy

<https://addons.mozilla.org/en-US/firefox/addon/foxyproxy-standard/>

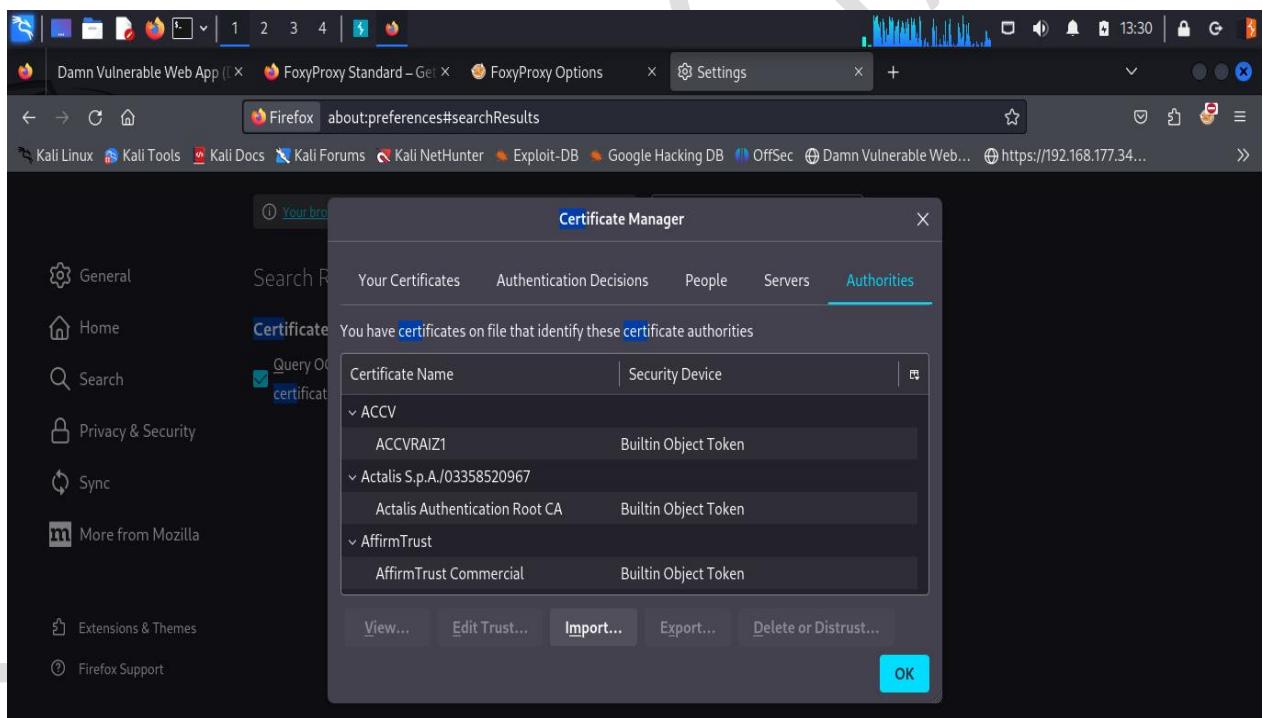


Step3: Go to new tab in browser, click on foxyproxy and click on options, click on proxies click on add button and fill the information like below

Advanced Cyber Security

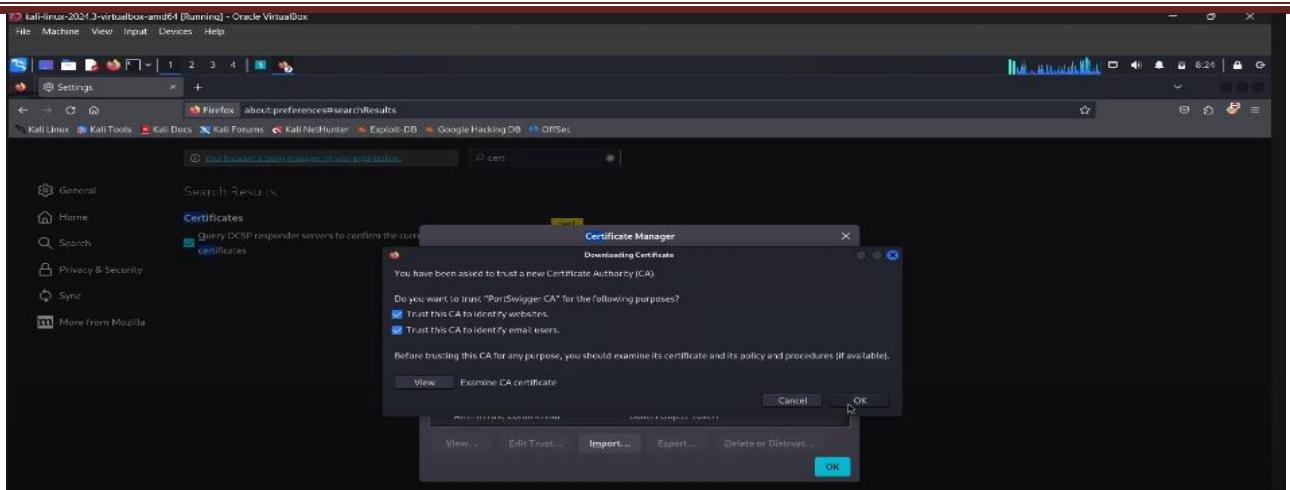


Step4: Open new tab and navigate to firefox setting, search for cert. Click on view certificate button
Then click on **import button** and select the downloaded certificate.

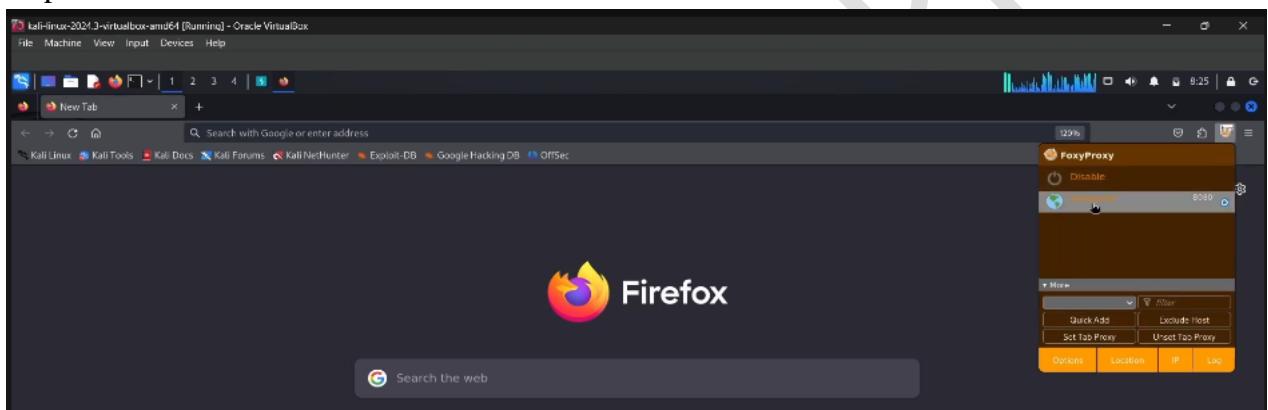


After that click on two checkboxes, now burpsuit is ready.

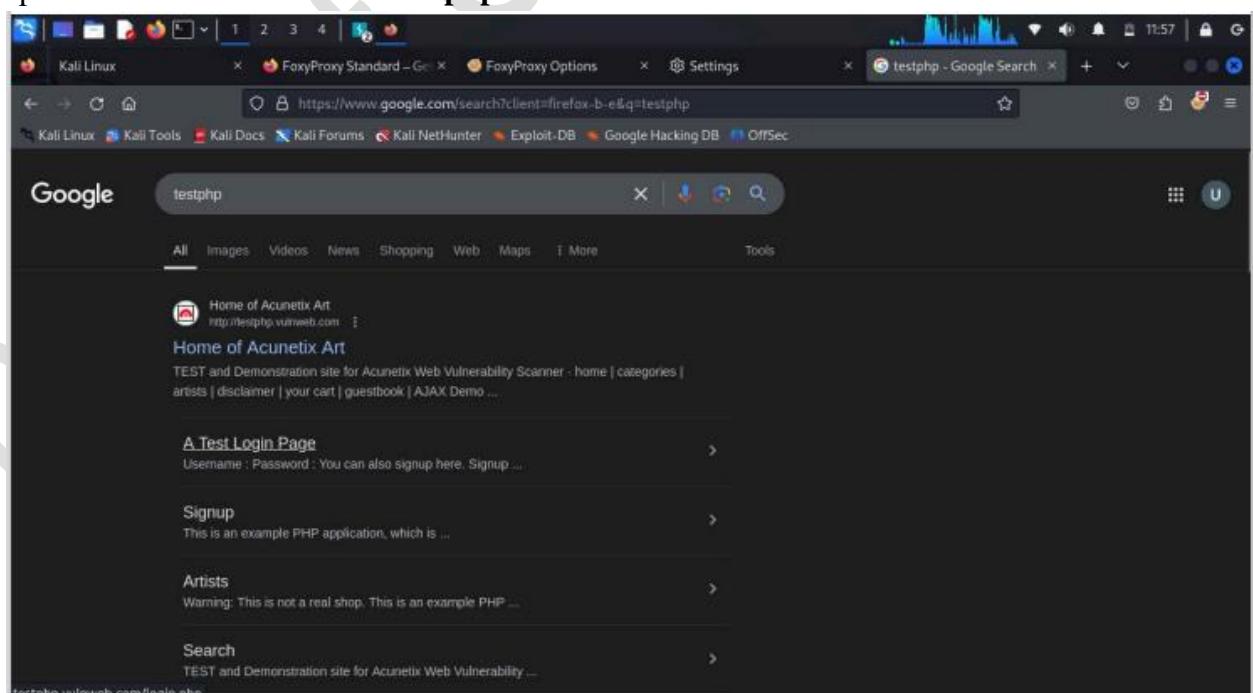
Advanced Cyber Security



Step5: Goto new tab and enable the extension



Step6: Goto browser search for Testphp



Advanced Cyber Security

Open the first link, before providing the credentials the burpsuit should be on



User name is **test** and password is **test**

The screenshot shows a web browser window with the following details:

- Address Bar:** testphp.vulnweb.com/login.php
- Page Content:** A login form with fields for Username and Password, both pre-filled with "test". Below the form, a note says "Signup disabled. Please use the username **test** and the password **test**".
- Sidebar (Left):** A sidebar titled "acunetix acuart" with a list of links:
 - TEST and Demonstration site for Acunetix Web Vulnerability Scanner
 - home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
 - search art go
 - Browse categories
 - Browse artists
 - Your cart
 - Signup
 - Your profile
 - Our guestbook
 - AJAX Demo
 - Links
 - Security art
 - PHP scanner
 - PHP vuln help
 - Fractal Explorer

The traffic captured by burp sent to **repeater**, then click on to forward/send. Next turn off the interceptor.

Advanced Cyber Security

The screenshot shows the Burp Suite interface with the Repeater tab selected. In the Request section, a POST request is shown to the endpoint /userinfo.php with the parameter ?uid=124. The response section displays the resulting HTML page.

```

1 POST /userinfo.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 20
9 Origin: http://testphp.vulnweb.com
10 Referer: http://testphp.vulnweb.com/login.php
11 Cookie: login=test%2Ftest
12 Upgrade-Insecure-Requests: 1
13
14 uname=test&pass=test
15

```

Step7: go to repeater and add code to first line with ?uid=124 ,then click on send.

The screenshot shows the Burp Suite interface with the Repeater tab selected. The POST request now includes the parameter ?uid=124 in the first line of the message. The response section shows the modified HTML page.

```

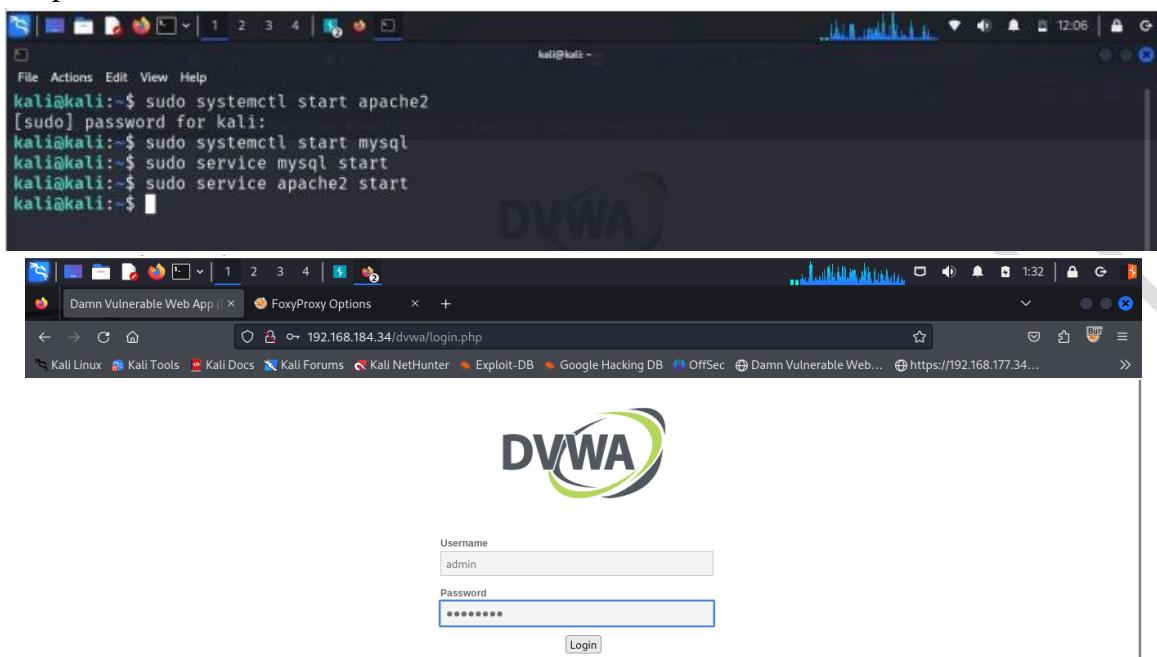
1 POST /userinfo.php?uid=124 HTTP/1.1
2 Host: testphp.vulnweb.com
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 20
9 Origin: http://testphp.vulnweb.com
10 Referer: http://testphp.vulnweb.com/login.php
11 Cookie: login=test%2Ftest
12 Upgrade-Insecure-Requests: 1
13
14 uname=test&pass=test
15

```

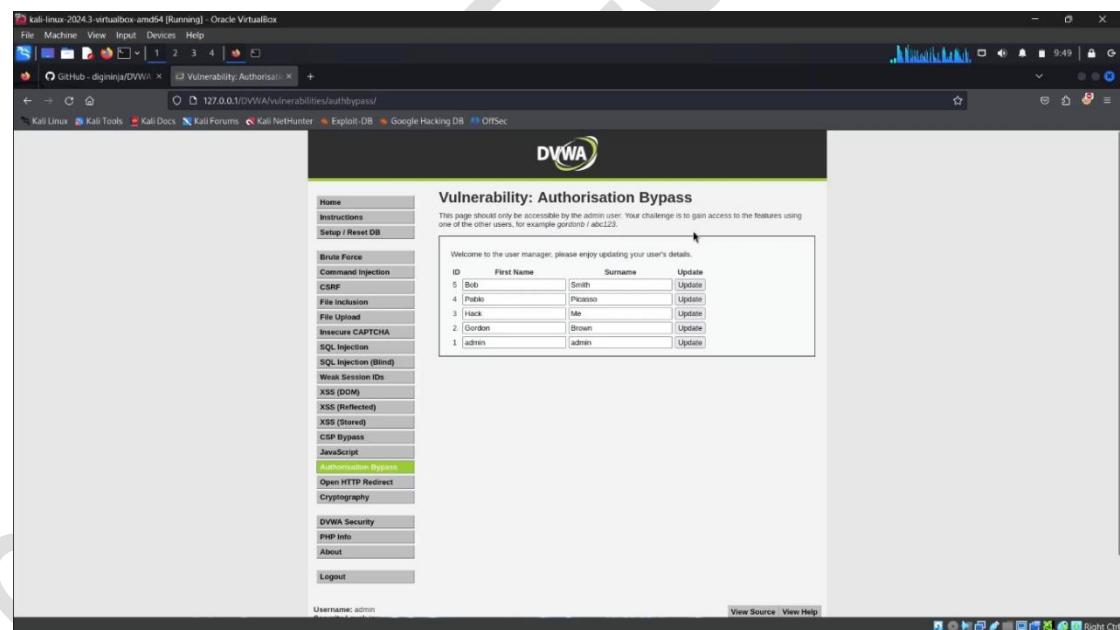
Now in response section click on show response in the browser option, then copy the link-burp suit is done.

The screenshot shows a web browser displaying the modified user info page from Acunetix. The URL in the address bar is testphp.vulnweb.com/userinfo.php?uid=124. The page content shows the user information form with the uid parameter set to 124.

Step 8: Open DVWA



We are at admin credentials:



Authentication Bypass → low level

Open up a new private window

Advanced Cyber Security

Vulnerability: Authorisation Bypass

Welcome to the user manager, please enjoy updating your user's details.

ID	First Name	Surname	Update
5	Bob	Smith	Update
4	Pablo	Picasso	Update
3	Hack	Me	Update
2	Gordon	Brown	Update
1	admin	admin	Update

Then navigate to local host DVWA 127.0.0.1 try to login user **gordonb** and password **abc123**
Copy the marked URL from **admin** and paste it into **new private window**

Vulnerability: Authorisation Bypass

Welcome to the user manager, please enjoy updating your user's details.

ID	First Name	Surname	Update
5	Bob	Smith	Update
4	Pablo	Picasso	Update
3	Hack	Me	Update
2	Gordon	Brown	Update
1	admin	admin	Update

New private window looks like, we **got admin access**

Advanced Cyber Security

The screenshot shows a web browser window titled "kali-linux-2024.3-virtualbox-amd64 [Running] - Oracle VirtualBox". The URL is "127.0.0.1/DVWA/vulnerabilities/authbypass/". The page displays a user management interface with a table of users:

ID	First Name	Surname	Update
5	Bob	Smith	Update
4	Dubbo	Picasso	Update
3	Hack	Me	Update
2	Gordon	Brown	Update
1	Admin	admin	Update

The sidebar on the left lists various DVWA vulnerabilities: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, JavaScript, Open HTTP Redirect, Cryptography, DVWA Security, PHP Info, About, and Logout.

Enumerate for the medium level , then it will show like unauthorized

The screenshot shows a web browser window titled "DVWA Security :: Damn V...". The URL is "127.0.0.1/DVWA/security.php". The page displays the DVWA logo and the title "DVWA Security".

Security Level

Security level is currently: **low**.

You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA:

- 1. Low - This security level is completely vulnerable and has no security measures at all. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
- 2. Medium - This setting is mainly to give an example to the user of bad security practices, where the developer has tried to fail to secure an application. It also acts as a challenge to users to refine their exploitation techniques.
- 3. High - This option is an extension to the medium difficulty, with a mixture of harder or alternative bad practices to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
- 4. Impossible - This level should be secure against all vulnerabilities. It is used to compare the vulnerable source code to the secure source code.

Prior to DVWA v1.9, this level was known as 'high'.

Medium

The sidebar on the left lists various DVWA vulnerabilities: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, JavaScripts, Authorisation Bypass, and Open HTTP Redirect.

At the bottom of the browser window, the status bar shows "Unauthorised".

Step9: Open burp suit, in browser turn on burpsuit

Advanced Cyber Security

The screenshot shows two windows side-by-side. On the left is a browser window for DVWA showing the 'Authorization Bypass' challenge. It displays a table of user details:

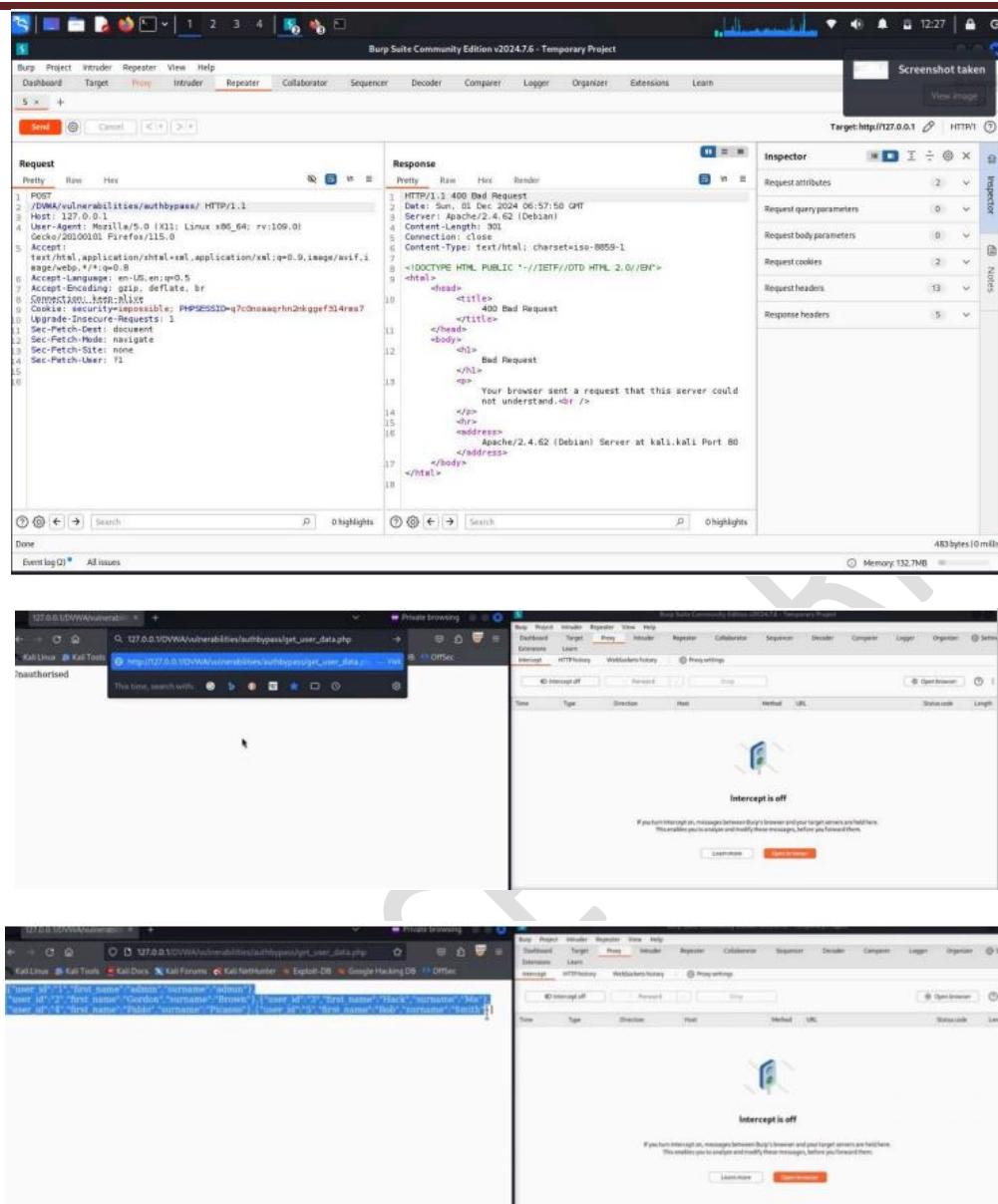
ID	First Name	Surname	Action
5	Bob	Smith	Update
4	Pablo	Picasso	Update
3	Hack	Me	Update
2	Gordon	Brown	Update
1	admin	admin	Update

The right window is Burp Suite showing the 'Proxy' tab. It has a message 'Intercept is off' and a note: 'If you turn Intercept on, messages between Burp's browser and your target servers are held here. This enables you to analyse and modify these messages, before you forward them.' There are 'Learn more' and 'Open browser' buttons.

Turn on the **interceptor** and refresh the unauthorized page

The screenshot shows the DVWA 'Unauthorized' page and the Burp Suite Request editor. In the Request editor, a context menu is open over the first line of the request. The 'Send to Intruder' option is highlighted. The request itself is a GET to the DVWA auth bypass page.

Advanced Cyber Security



Link: https://www.youtube.com/watch?v=0G08ydvmmYY&ab_channel=MaCT