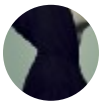
# Stories Of IDOR



Shivbihari Pandey Follow
Sep 28 · 4 min read

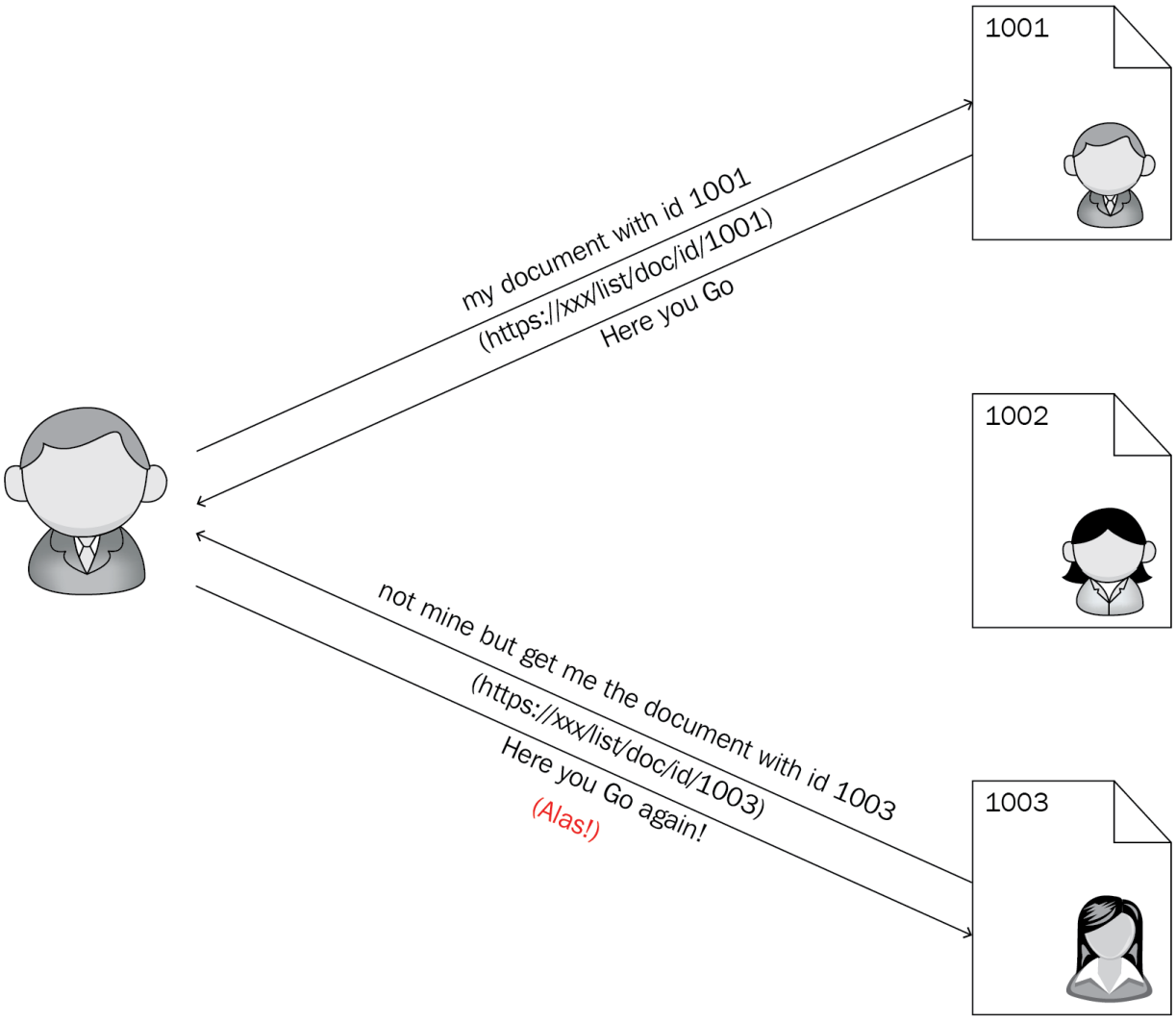
Hello

Welcome Back ,

This is going to be Series ,where Iwill Share My Findings .

> ***What Is IDOR:***

Insecure Direct Object References occur when an application provides ***direct access to objects*** based on user-supplied input. As a result of this vulnerability attackers can ***bypass authorization and access resources in the system directly.***




in simple language , suppose there is 2 user account , U1 & U2,

and both have files in there accounts, but only account user can access it,means U1 can only access his account files not U2 Files.

one day U1 trying to view his file blahBlah.pdf , file url ,in browser look like :

*https://whocare.com/file/23*

Now Curious U1 try to change the last Number, try to see what happen, like

*https://whocare.com/file/50*

Now he able to view U2 file from his account.

Now questions is why:

Because Application provide ***Direct Access to Object based on user input*** and without ***validating the authenticity of objec***t.

### *How to Find it*

Well IDOR is present in application like XSS, it will very easy to find it, but it become easier after you understand the purpose/workflow of application you testing.

I am going to share some of my finding, which will clear your concept, how and where to find these issues.

### *Part 1:* IDOR Can Able To view Other User Account Details

This is begin while i was Signing UP the User account , Request got intercepted by the Burp Proxy is look like something this:

whocare.com Domain make an Internal API call , for Sign-Up.

```
POST /analytics/index.php/plus/registerportal?user_id=43657&key=344246b382b1d70c2
Content-Type: application/x-www-form-urlencoded
Content-Length: 239
Host: api.whocare.com
Connection: close
Accept-Encoding: gzip, deflate
User-Agent: okhttp/3.4.1

email=test@whocare.com&password=as&username=&firstname=tvbb&lastname=gvcz&mobile=
```

**re** hosted with ❤ by **GitHub** **view raw**

Request

Now if you see in request there is Parameter ***user_id*** , for testing purpose changed it to random values, and i got response as an error like : ***user is already existed ,*** along with that it disclosed the User Information like : **Name, Email, Address** etc etc.

```
HTTP/1.1 200 OK
Date: Tue, 05 Mar 2019 12:53:30 GMT
Content-Type: text/html
Connection: close
Set-Cookie: __cfduid=dc4d54bb59b5c4a2c8501e3ed1cd5952b1551790409; expires=Wed, 0
Vary: Accept-Encoding
Server: cloudflare
CF-RAY: 4b2c3badddb3ce21-LHR
Content-Length: 311

{"error":"exist","user":[{"id":"34666","username":"admin","firstname":"Pappu","l
```

**Response** hosted with ❤ by **GitHub** **view raw**

Response

Well i redacted some of the Personal information, because i got the Admin account details, Actually this was not come bup in one shot, for this i started to Brute force the ***user_id*** parameter Using Burp Suite Intruder, i got many users details, in which i found the Admin account detail too.

> ***Part-2: IDOR : Can Unsubscribe Anyone User Email From Subscription list***

In same website whocare.com [DummyName], there is option for subscribe for newsletter , for Email Notification for latest Updates.

In User Account Setting there is option for the Unsubscribe from Newsletter, when you submit the request, they will send an email to registered users,and URL look like this:

http://whocare.com/deleteNewsletter
/dmljdGltZW1haWxAZ21haWwuY29t

If you see, it's base64 Encoding ,

***dGVzdGVybWFpbEBnbWFpbC5jb20= : testermail@gmail.com***

Now we need Users Email In Order to Unsubscribe User From Newsletter, because they are not Validating this request.

Now From First issue we able to get the user information like Email, now you know if you want to unsubscribe all the users from website you just need email-address of the user, which you have. for attack ,intercept this url request, send it to Intruder and add all the emails of users and make an base64 encode before submitting, start attack . Period

So i was trying to chain the 2 small IDOR into Impactful Report.

> ***Part-3: Open Mail Relay Identified: Can Send Spoof Email To Victim From Authentic Whocare.com Mail Server***

this is another Vulnerability Exit in Same domain, in ***Feedback section***

From where you can submit feedback to Team .

request for this look like:

```
POST /Services/PostContactUsEmail HTTP/1.1
Host: www.whocare.com
Connection: close
Content-Length: 327
Accept: */*
Origin: https://www.whocare.com
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Referer: https://www.whocare.com/feedback
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: [REDUCTED]

ContactUs_Name_Txt=hiname&ContactUs_Email_Txt=attacker@gmail.com&ContactUs_Messa
```

**Feedback** hosted with ♥ by **GitHub** view raw

Now if you See in the Request Section there is 2 Parameter we will use for attacking purpose:

***ContactUs_Department_Txt=*** account where email to be send

***ContactUs_Email_Txt=*** account from where email send

***ContactUs_MessageBody_Txt***= Message you like to send

now i can craft New Request and change parameters with like this:

***ContactUs_Department_Txt=admin@whocare.com***

***ContactUs_Email_Txt=[Use All the Users List For Attacking***

***purpose]***

now all the user will get the email from admin account , which look legitimate, a perfect attack for phishing.

I have other Stories About IDOR, hopefully i will write about those in future.

## Remediation:

A proper access control need to be implemented, means these requests should be validated before it proceeds, another thing is always

Use strong and random encryption instead of numbers. Like id=3 , instead of 3 ,use some random encryption.

For more information please visit

https://cheatsheetseries.owasp.org/cheatsheets /Insecure_Direct_Object_Reference_Prevention_Cheat_Sheet.html

> ***Timeline:***

1. Report Send
2. Get Patched

3. Bounty Awarded[Whocares 🤷‍♀ except me]

That's it for now, we will meet soon with our next Blog. Till then Goodbye.

If you Like this post, feel free to retweet.

. . .

*Follow Infosec Write-ups for more such awesome write-ups.*