



# IDOR Bug Hunting Methodology

Become a Successful  
Bug Bounty Hunter



## IDOR – Basics

- Insecure Direct Object Reference
- You can create, read, update, delete information of resources you should not be able to
- Missing authorization checks
- Example: Reading someone else's shipping address



## IDOR – Example

- How to find it?
- Look for ID parameters like  
`?users/user_id=1234`
- Look for other IDs as well like names
- `?users/user_id=johnsmith`





# IDOR – Example

Example

[https://www.example.com/order?order\\_id=1337](https://www.example.com/order?order_id=1337)

Test for IDOR

[https://www.example.com/order?order\\_id=1336](https://www.example.com/order?order_id=1336)



# IDOR – How to find them?

## Example with complex GUID

[https://www.example.com/order?order\\_id=425d1126-4139-4067-ac68-d9caafdf2b46](https://www.example.com/order?order_id=425d1126-4139-4067-ac68-d9caafdf2b46)

### What do you do?

- Brute forcing a dead end
- Generate 10 – 20 samples (any patterns?)
- Look for GUIDs in other parts of the app!
- Maybe they leak when updating something
- Google the GUID
- Sometime IDs are base64 encoded
- Still try Integers (like 1, 2, 3) – backend may process it the same way!



## IDOR – Where?

- IDORs in Web Applications
- IDORs in Mobile Applications
- IDORs in APIs all the time
- Not only think of IDs but also or other values like username, email, mobile number, etc.
- IDORs between levels (unauth, guest, user, moderator, admin etc.)



## IDOR – JSON PUT and POST

Anytime you see a request and the postdata is JSON (POST or PUT):

```
{"userid":"1", "password":"oops"}
```

Try to modify the userid:

```
{"userid":"2", "password":"oops"}
```





## IDOR – Inject Parameters

Inject a new parameter like a userid even though the original request doesn't contain it. Again look for JSON (POST or PUT):

Original POST  
`{"password":"oops"}`

Modified POST  
`{"userid":"1", "password":"oops"}`





## IDOR – CRUD Methods

- IDORs are not just about reading (GET)
- But also about creating (POST)
- Updating (PUT)
- Deleting (DELETE)



## IDOR – Enumeration

Enumerating with Burp at scale

<https://api.example.com/api/user/139349>

GID: 139349

enumerate

<https://api.example.com/api/user/x>



## IDOR – Common Places

- Other common places for IDORs
- Opt in / Opt out links
- Mobile Apps
- Updating account settings  
(look for PUT and POST requests)
- Reset password





## IDOR – Logic Flaws

- Anytime you see some identifier
- This can be a simple ID (1), or a GUID (425d1126-4139-4067-ac68-d9caafdf2b46),
- On large UUIDs they may reuse a set of characters and only change a few
- Sometimes when displaying the image this value may be used in the URL, <https://cdn-images.example.com/avatar/ac68-d9caafdf2b46/0.jpg>



Thank You!

Become a Successful  
Bug Bounty Hunter