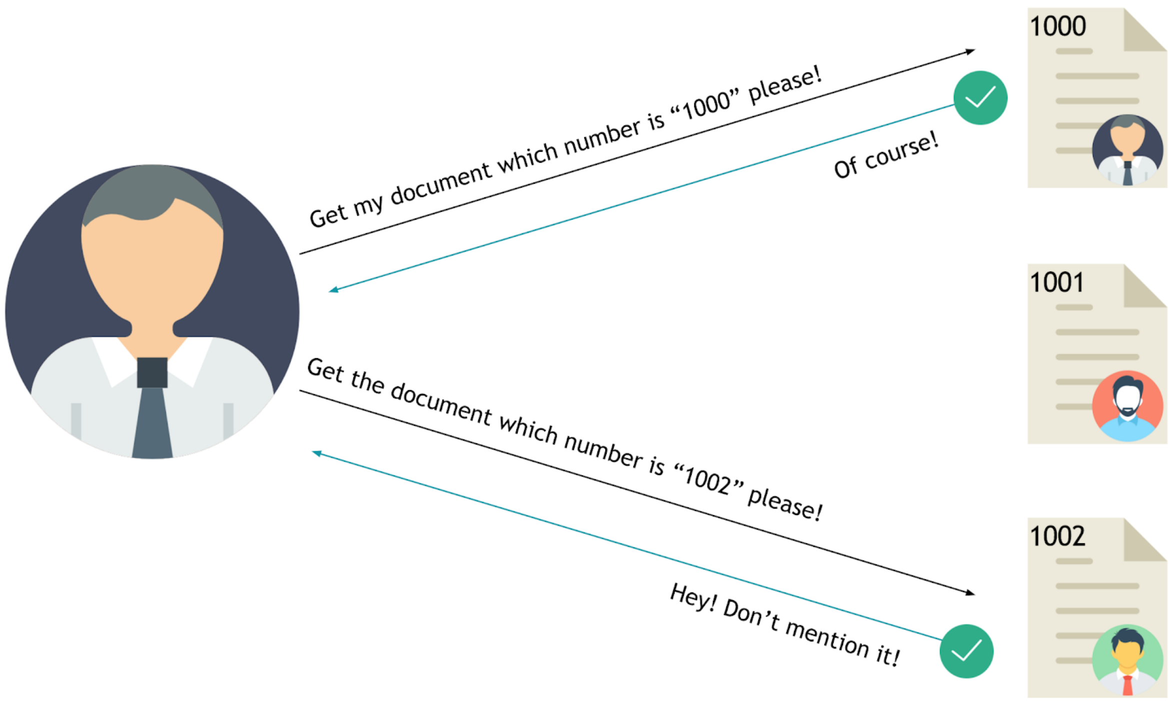
💻

IDOR: Insecure direct object reference

What is IDOR?

An insecure direct object reference (IDOR) is an access control vulnerability where invalidated user input can be used for unauthorized access to resources or operations. It occurs when an attacker gains direct access by using user-supplied input to an object that has no authorization to access. Attackers can bypass the authorization mechanism to access resources in the system directly by exploiting this vulnerability.

Every resource instance can be called as an object and often, represented with and ID. And if these IDs are easy enough to guess or an object can be used by an attacker to bypass access check somehow, we can talk about an IDOR at this point.


Get my document which number is "1000" please!
Of course!
Get the document which number is "1002" please!
Hey! Don't mention it!
1000
1001
1002

Referring to the above image, an attacker by ethical means can only get the document numbered `1000` which is legally his. But what if the web application does not validate the `number` and an attacker puts the `number` of its victim. This can cause the attacker get hold of the sensitive document which he should not have access to.

Exploiting IDOR's

Exploiting IDOR's is very simple. You just need to find an entry point where in you can change the `ID` in order to authenticate as another user and check out its details.

The steps to exploit this vulnerability are:

- Find an entry point.
- Change the value of that parameter to something else
- Send the request and check if you have been authenticated or have got the resource that does not belong to you.

Severity

The severity of IDOR varies from P3 to P2 depending on what data is being exposed.

Impact of IDOR

This vulnerability violates the privacy of a User. Without authentication or access limits, an attacker could easily build a program to download every post, photo, video, and data from the entire site. While this was just public posts (not necessarily IDs used to verify accounts), geolocation data from posts was also downloaded, which could reveal GPS coordinates of users' homes.

Prevention of IDOR

To remediate IDOR vulnerabilities, below are a few best practices.

- Developers should avoid displaying private object references such as keys or file names.
- Validation of parameters should be properly implemented.
- Verification of all the referenced objects should be checked.
- Tokens should be generated in such a way that it can only be mapped to the user and is not public.
- Ensure that queries are scoped to the owner of the resource.
- Avoid things like using UUIDs (Universally unique identifier) over Sequential IDs as UUIDs often let IDOR vulnerabilities go undetected.

References

- IDOR by Port Swigger: https://portswigger.net/web-security/access-control/idor
- IDOR by OWASP: https://cheatsheetseries.owasp.org/cheatsheets/Insecure_Direct_Object_Reference_Prevention_Cheat_Sheet.html
- How to find IDOR's by Bugcrowd: https://www.bugcrowd.com/blog/how-to-find-idor-insecure-direct-object-reference-vulnerabilities-for-large-bounty-rewards/

Lab Documentation