How to Test Your App for IDOR

IDOR is when changing an ID in a link shows you someone else's data. Here's how to test your app for it and fix it.

Your app checks that you're logged in. It never checks that the thing you asked for is yours.

1 Let your AI do it

Paste this into an AI coding tool that can run commands (for example Claude Code, Cursor or Codex) inside your project. It finds every place that takes an ID, proves each hole with two test accounts, asks before fixing, then re-tests.

COPY THIS INTO YOUR AI CODING TOOL

```
Read https://github.com/vxsec/idor-check and follow its AGENTS.md to check this app for IDOR. Show me proof for anything you find, and fix it after I say yes.
```

2 Do it yourself

Five minutes, two accounts, nothing to install.

1. **Make two accounts, A and B.**
   Use two browsers, or a normal and a private window.
2. **As B, create something private.**
   An order, a note, an upload. Copy its link.
3. **As A, open B's link.**
   If you see B's data, you've found IDOR.

4 No ID in the link?

Press F12, open Network, and find a request with an ID in it. As A, right-click it, Copy as fetch, paste it into Console, swap in B's ID and press Enter.

5 Try edits and deletes the same way.

Then log out and try again.

PASS
You get "not found" or "not allowed" every single time.

FAIL
You see, change or delete B's data even once.

Found a hole by hand? Paste this into your AI tool:

COPY THIS INTO YOUR AI TOOL

```
Go through every route, API endpoint, server action and database query in
this app that takes an ID from the user. Make sure the server checks that
the logged-in user is allowed to read, update or delete that record. Never
trust an ID or user id sent from the browser. If this app uses Supabase:
turn on Row Level Security for every table, drop policies that allow all
rows (like using (true)) unless that table is meant to be public, and give
each private table only the policies the app needs (select, insert, update
or delete), each tied to auth.uid(). Don't add permissions the app doesn't
use. Keep anything that's meant to be public or shared working. Then write a
test where user A tries to read, edit and delete user B's records, and to
create a record with B's user id, and make it pass.
```

Then test again yourself. Don't take the AI's word for it.

Only test apps you own or have written permission to test.

Built by VXSec · vaultxsec.com