Express.js Cookie Explanation

This snippet sets a cookie called "refreshToken" in an Express.js application:

res.cookie("refreshToken", refreshToken, {

httpOnly: true,

secure: false, // dev -> false, prod -> true

sameSite: "lax",

path: "/",

});

Explanation of each option:

1. httpOnly: true

- Cookie cannot be accessed via JavaScript (document.cookie).

- Helps prevent XSS attacks.

- Only the browser automatically sends it back to the server.

2. secure: false

- If true, cookie is only sent over HTTPS.

- In development (localhost, no HTTPS), keep it false.

- In production, always set it to true.

3. sameSite: "lax"

- Controls cross-site cookie behavior.

- "lax" means cookies are sent for top-level navigation (GET) but blocked in some cross-site requests (e.g., POST/iframe).

- Helps mitigate CSRF attacks.

4. path: "/"

- Defines the scope of the cookie.

- "/" means the cookie is valid across the entire site.

Summary:

This configuration securely stores a refresh token inside an HttpOnly cookie.

- Development: secure = false

- Production: secure = true, always use HTTPS