Express 8

Signed Cookies

Introduction

Cookies are small pieces of data stored on the client-side, commonly used for maintaining user sessions, remembering user preferences, and tracking user behavior. Express.js, a popular web application framework for Node.js, provides built-in support for handling cookies.

However, **regular cookies can be easily tampered with on the client-side**, potentially leading to security vulnerabilities. This is where signed cookies come into play, offering an additional layer of security and data integrity.

What are signed cookies ?

Signed cookies are regular cookies that have been cryptographically signed. This means that while the content of the cookie is still visible to the client, any attempt to modify the cookie will be detected by the server.

The main purpose of signed cookies is to ensure data integrity. They allow the server to verify that the cookie content hasn't been tampered with since it was set by the server.

Unlike encrypted cookies, which hide the content, signed cookies keep the data visible but make it tamper-evident.

How signed cookies work ?

The signing process involves creating a hash of the cookie's content along with a secret key known only to the server. This hash is then appended to the cookie value.

When the cookie is sent back to the server, it recalculates the hash using the received cookie value and its secret key. If the calculated hash matches the one in the cookie, it verifies that the cookie hasn't been tampered with.

The secret key plays a crucial role in this process. It should be a long, random string that is kept secure on the server.

Implementing signed cookies in Express.js - Step 1

1. Install the cookie-parser module
2. Import the cookieParser middleware from cookie-parser
3. Use the cookieParser middleware and pass a secret that will be used to create and read signed cookies.

```
const express = require('express');
const cookieParser = require('cookie-parser');

const app = express();
const secretKey = 'your-secret-key';

app.use(cookieParser(secretKey));
```

Implementing signed cookies in Express.js - Step 2

Setting a signed cookie

```
app.get('/set-cookie', (req, res) => {
  res.cookie('user', 'john_doe', { signed: true });
  res.send('Signed cookie set');
});
```

Implementing signed cookies in Express.js - Step 3

Reading a signed cookie

```
app.get('/get-cookie', (req, res) => {
  const user = req.signedCookies.user;
  res.send(`User from signed cookie: ${user}`);
});
```

Handling Tampered Cookies

```
app.get('/verify-cookie', (req, res) => {
  const user = req.signedCookies.user;
  if (user === undefined) {
    res.send('Cookie was tampered with or doesn\'t exist');
  } else {
    res.send(`Valid signed cookie: ${user}`);
  }
});
```