

Programação na Internet

Turma i52d

Lesson 40 and 41

Cookies, Session and Authentication (with passportjs)

Authentication

- Goal: get the user identity!

1x: Credentials ---> User ----> User ID (on session state)

Further requests: User ID ----> User (on Web App)

=> HTTP is stateless => How can we keep User ID on session state?

R: Some how with Cookies!!!!

Cookies

A cookie is **only** a header in an HTTP request or response!

Special behavior with the browser.

1. A cookies born in the server and is sent to the browser.
2. The browser sends back the cookie to the origin.

By default:

- Life cycle is session, i.e. browser life cycle.

Cookies in action...

Let's see cookies in action:

- Using cookie-parser module:
- `require('cookie-parser')()`
 - => returns a middleware (express handler)
- `app.use(require('cookie-parser')())` => intercepts all HTTP requests:
 - req – Adds a property: `cookies`
 - res – Adds a method: `cookie(name, value, options)`
 - Options has the values to append on header `Set-Cookie`

Session rather than Cookies only

- With session we maintain one only cookie with the Session Id.
- Session maintains data in server associated to each Session Id.

HTTP request ----> session ID ----> server restores data from Session Id

++ Less cookies

--- By default, session has no persistence!

Session in action...

- `require('express-session')(...)`
 - => returns a middleware (express handler)
- `app.use(require('express-session')(...))` => intercepts all HTTP requests
 - => provides an object in `req.session`
 - => `req.session` is kept in server
 - => maintains a single cookie `connect.sid` per session

Authentication

(with passportjs)

Authentication

- Goal: get the user identity!

1x: Credentials ---> User ----> User ID (on session state)

Further requests: User ID ----> User (on Web App)

Passport

- Based in different and **unlimited** authentication strategies:
 - OAuth
 - Local
 - Etc...

+++ Flexible

---- Complex

WE ARE NOT GOING TO USE STRATEGIES

Login error

`res.redirect('/vinyl/login')` + send the Error message

Problem: the Error message should be attached with redirect and next GET

Solution ?

R: keep it on session

R: Remember to clean message when arrive to /login

i.e. delete it on first read.

=> Already provided by flash libraries (e.g. connect-flash)