Devoir 1 / Module 2

1/ ﻿A CSRF middleware was not detected in your express

Application

Ajout import modules :

﻿const csurf = require('csurf');

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

Puis activation protection :

app.use(cookieParser());

app.use(csurf({ cookie: true }));

2/ Path transversal

Ajout const fs = require('fs'); dans les imports modules

Ajout : const safeFilename = path.resolve(filename);

Et utilisation de safeFilename au lieu de filename

+ const safeFilename = filename.replace(/(\.\.\/|\.\.|\/)/g, '');

Et if (!normalizedPath.startsWith(path.join(\_\_dirname, 'uploads'))) { return res.status(403).send('Accès interdit');

Pour securiser :

﻿const safeName = he.encode(name);

const safeAge = he.encode(age.toString());

r﻿es.render('templateName', { name: he.encode(name), age: he.encode(age) });

3/ ﻿Detected user input used to manually construct a SQL

String

Utilisation dangereuse de $user dans la requete SQL

Modification :

const query = `SELECT \* FROM users WHERE name = ?`;

db.all(query, [userName], (err, rows) => {…

Utilisation directe de name et age dans la requete SQL

Modification :

Ajout const query = `INSERT INTO users (name, age) VALUES (?, ?)`;

Puis correction:

db.run(query, [name, age], function(err) { // Passer les valeurs comme paramètres if (err) { res.status(500).send('Erreur lors de l\'insertion'); } else { res.send(`Bonjour ${name}, vous avez ${age} ans.`); } }); });

4/ ﻿Detected directly writing to a Response object from user-

defined input

name et age sont directement inserés dans la réponse

ajout de la bibliothe que he : he.encode(name) he.encode age

+ if (typeof name !== 'string' || name.length === 0 || typeof age !== 'number') { return res.status(400).send('Données invalides'); }