



Universidad Veracruzana

## **Licenciatura en Ingeniería de Software**

Facultad de Estadística e Informática

### **CWE-209: Generation of Error Message**

#### **Containing Sensitive Information**

Programación Segura (12800)

Guillermo Humberto Vera Amaro

**Daniel Mongeote Tlachy**

Xalapa, Ver; martes 02 de abril del 2024

**Instrucción.** Elaborar una aplicación con Node.js que permita ejemplificar la vulnerabilidad Desreferencia de puntero NULL en una aplicación web cliente-servidor.

Coloca las siguientes capturas de pantalla, cada una con una descripción textual arriba de la captura.

- Coloca la captura de pantalla de tu código fuente en Visual Studio Code del archivo app.js.

```
1 // Click here to ask Blackbox to help you code faster
2 const express = require('express')
3 const request = require('request')
4 var session = require('express-session')
5 const app = express()
6 app.use(session({ secret: 'miappsegura', resave: false, saveUninitialized: true }))
7 const port = 3000
8 //app.use(express.static('public'))
9 app.use(express.urlencoded({ extended: true }))
10 app.set('view engine', 'ejs')
11 let api = {
12   // Esta key nos da acceso a toda la API. No compartas esta llave con nadie!
13   key: "88665751-288d-4175-852f-6519d79fd1f"
14 }
15 app.get('/', (req, res) => {
16   req.session.destroy()
17   res.render('index')
18 })
19 app.get('/welcome', (req, res) => {
20   if (req.session.username) {
21     res.render('welcome', { username: req.session.username })
22   } else {
23     res.send('No esta autorizado para ver este contenido.')
24   }
25 })
26 app.post('/', (req, res) => {
27   // Obtenemos el dominio en el que el usuario nos visita
28   // pues tenemos varios en el cluster
29   let host = req.get('host')
30   let username = req.body.username
31   let password = req.body.password
32   // Verificamos que este operacional
33   if (request(`http://${host}/codigos?api_key=${api.key}`)) {
34     console.log('Funcionando')
35   } else {
36     console.log('No esta funcionando la API')
37     return res.render('index', { username: 'username' })
38   }
39   // Contactamos a nuestra API para obtener los usuarios
40   request(`http://${host}/codigos?api_key=${api.key}`, function (error, response, body) {
41     let users = JSON.parse(body)
42     var user = users.find(u => u.name === username && u.code === password); // encuentra el usuario.
43     if (user) {
44       req.session.username = username
45       res.redirect('/welcome')
46     } else {
47       res.render('index', { username: 'username' })
48     }
49   })
50 })
51 // Cuidado con esto, solo se puede ver con la api-key
52 app.get('/codigos', (req, res) => {
53   // validamos que tenga permiso de leer esto
54   if (req.query.api_key === api.key) {
55     res.json(codigos)
56   } else {
57     res.status(401).send('Sin autorización')
58   }
59 })
60 app.listen(port, () => {
61   console.log('Aplicación de ejemplo escuchando en el puerto ${port}')
62 })
63 var codigos = [
64   { name: 'admin', code: "patito" },
65   { name: 'user', code: "34fiufde0q5" },
66   { name: 'guest', code: "uyy8787##$%K" }
67 ]
```

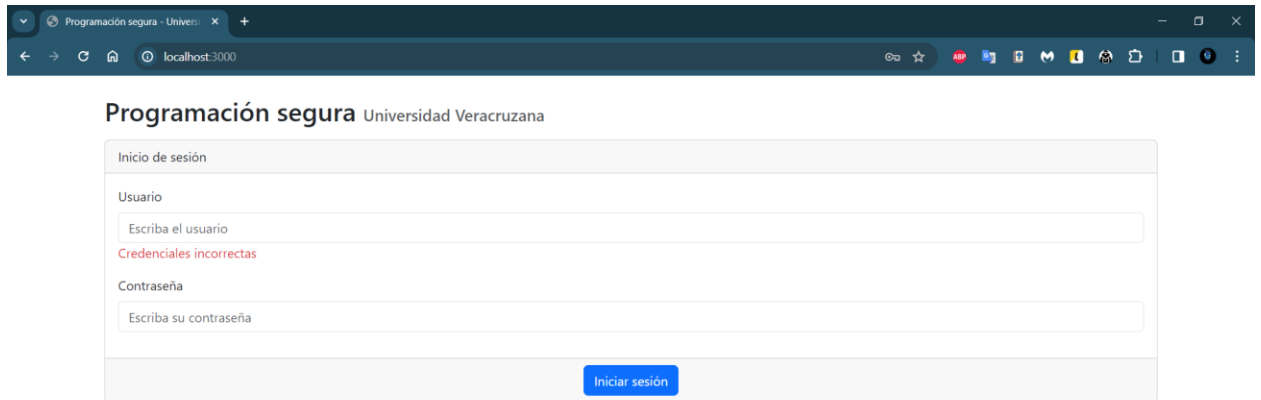
- b. Coloca la captura de pantalla de la salida de la página index.html de autenticación.

The screenshot shows a web browser window with the address bar displaying 'localhost:3000'. The page title is 'Programación segura Universidad Veracruzana'. The main content area contains a login form with the following elements:

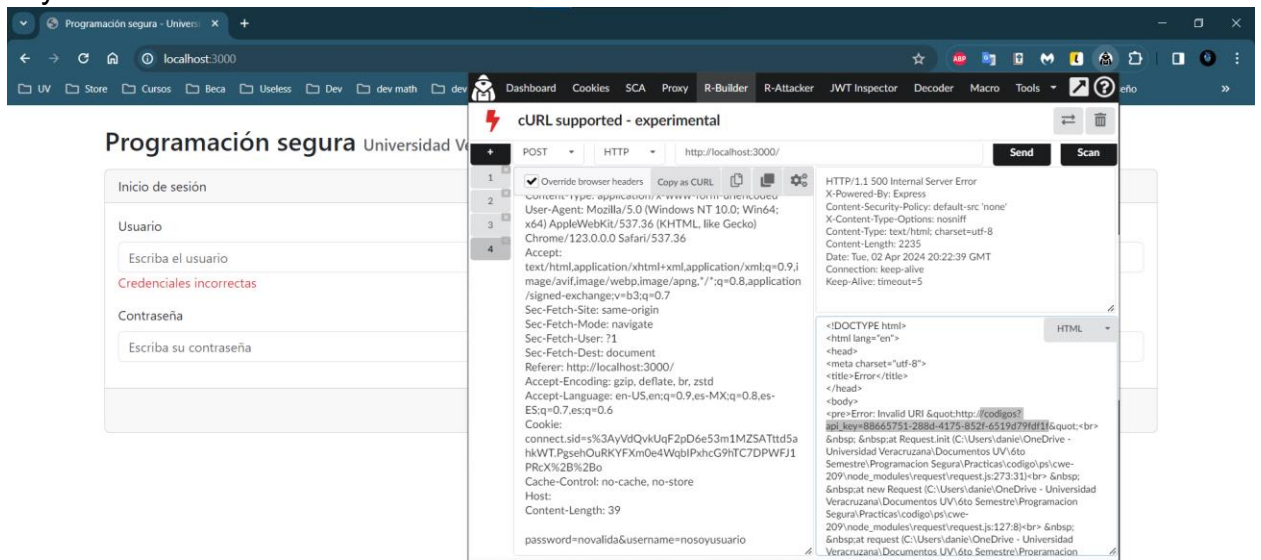
- A header section titled 'Inicio de sesión'.
- A label 'Usuario' above a text input field with the placeholder text 'Escriba el usuario'.
- A label 'Contraseña' above a text input field with the placeholder text 'Escriba su contraseña'.
- A blue button labeled 'Iniciar sesión' at the bottom right of the form.

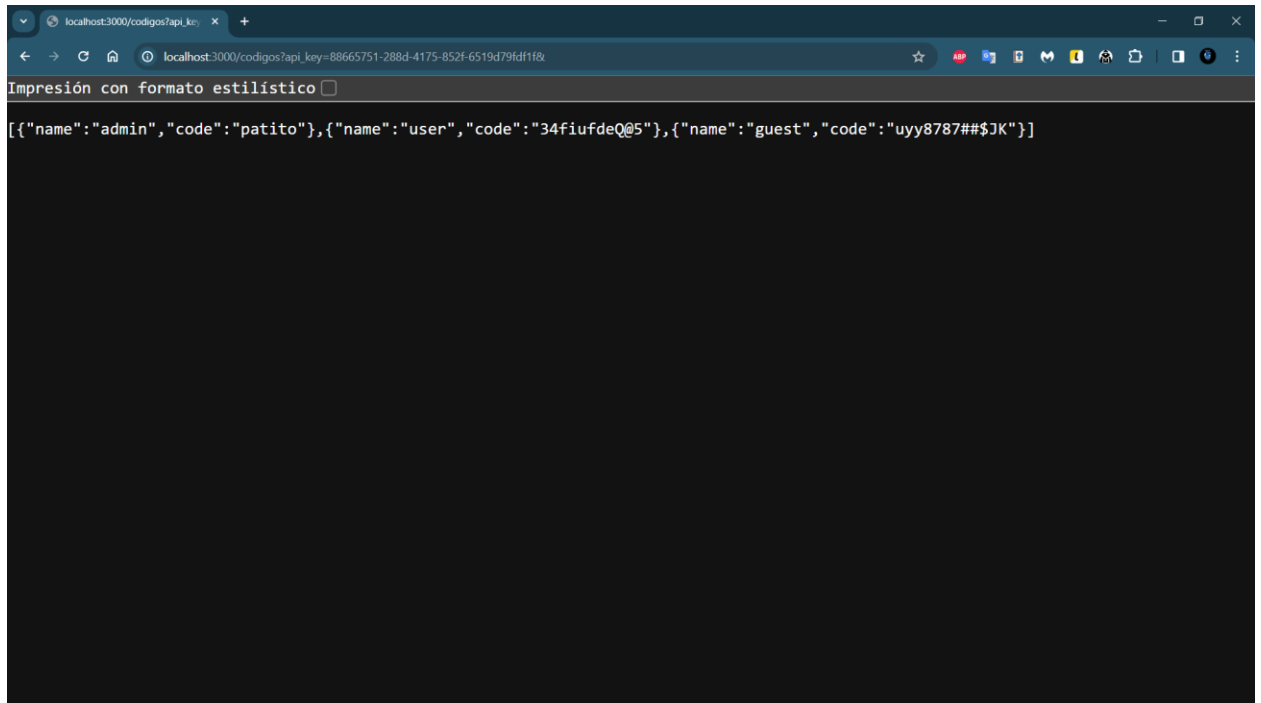
The screenshot shows a web browser window with the address bar displaying 'localhost:3000/welcome'. The page title is 'Programación segura Universidad Veracruzana'. The main content area contains a protected site message with the following elements:

- A header section titled 'Sitio protegido'.
- A message: 'Bienvenido **admin** a este sitio protegido por correo electrónico y un código de acceso.'
- A button labeled 'Cerrar sesión'.



- c. Coloca la captura de pantalla de la salida del error de Node.js donde se vea la api key revelada al atacante.





- d. Publica tu código fuente en un proyecto público de GitHub. Coloca la URL del código fuente publicado en GitHub.

URL: <https://github.com/danieltlachy/CWE-209-Generation-of-Error-Message-Containing-Sensitive-Information.git>