



# **UNIVERSIDAD DE LAS FUERZAS ARMADAS “ESPE”**



## **CARRERA DE INGENIERÍA DE SOFTWARE**

### **Integrantes:**

Carlos Romero, Jimena Tutillo

### **Asignatura:**

Desarrollo de software Seguro

### **Docente:**

Diego Medardo Saavedra Garcia

**NRC:** 15594

### **Período:**

Pregrado S-II MAY24 – SEP24

**Fecha:** 27-Jul-2024

# **DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

## **GUIA DE PRACTICA DE LABORATORIO /TALLER**

### **1. TEMA:**

Proyecto Integral de Seguridad Informática.

### **2. OBJETIVOS**

#### **2.1. OBJETIVO GENERAL**

El objetivo de esta actividad es aplicar y consolidar los conocimientos adquiridos a través de un proyecto integral. Los estudiantes diseñarán, implementarán y evaluarán la seguridad de un sistema web utilizando herramientas y técnicas, abordando aspectos de seguridad ofensiva y defensiva, así como la implementación de mejores prácticas de seguridad.

### **3. INTRODUCCIÓN**

La seguridad informática se ha convertido en un componente crítico para el desarrollo y mantenimiento de aplicaciones web, dado el creciente número de amenazas y ataques cibernéticos. Un Proyecto Integral de Seguridad Informática tiene como objetivo proteger los sistemas de información y los datos sensibles contra accesos no autorizados, uso indebido, modificación o destrucción. Este tipo de proyecto abarca varias disciplinas y técnicas, incluyendo la identificación y mitigación de vulnerabilidades comunes como el SQL Injection y el Cross-Site Scripting (XSS).

### **4. DESARROLLO:**

#### **4.1. Diseño del Sistema Web**

Sistema web que permite la gestión eficiente de equipos de mecánica mediante una interfaz intuitiva. La aplicación implementará un sistema CRUD (Crear, Leer, Actualizar, Eliminar) para facilitar el manejo de los datos de los equipos, así como un sistema de autenticación para asegurar el acceso a la información.

*Características Principales:*

Autenticación de Usuarios: Registro de nuevos usuarios con validación de datos.

Gestión de Equipos:

- Crear: Permitir a los usuarios agregar nuevos equipos con detalles como nombre, tipo, estado, y ubicación.
- Leer: Visualizar la lista de equipos registrados, con opciones de búsqueda y filtrado.
- Actualizar: Modificar la información de los equipos existentes.
- Eliminar: Eliminar equipos que ya no sean necesarios.

# DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

## GUIA DE PRACTICA DE LABORATORIO /TALLER

Base de Datos: Implementación de una base de datos para almacenar la información de los usuarios y los equipos.

Seguridad:

- Protección de datos sensibles mediante encriptación.
- Validación de datos en el lado del servidor para prevenir inyecciones SQL y XSS.

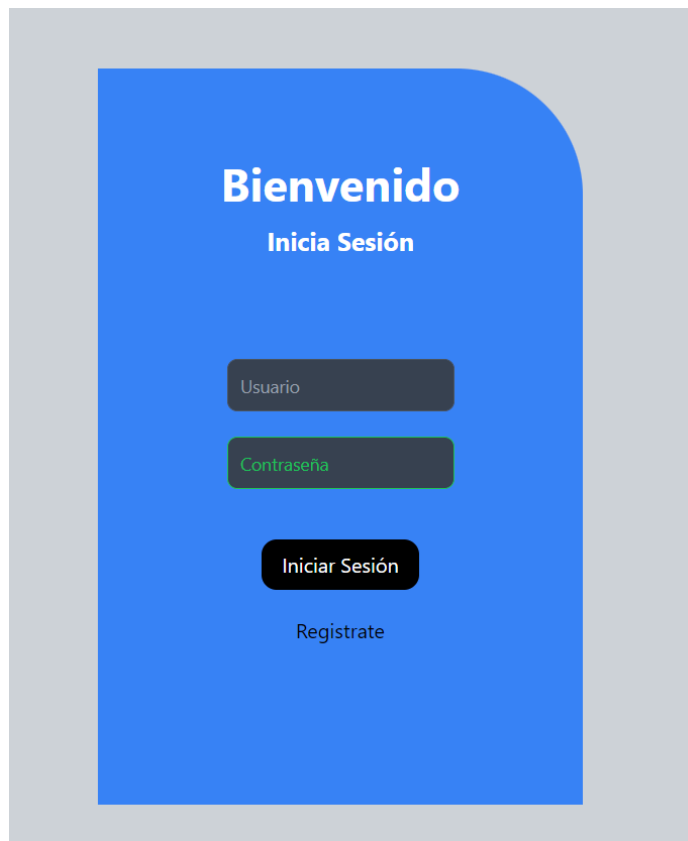
### 4.2. Implementación del Sistema Web

**Descripción de framework:**

**Frontend:** React (Javascript) + Vite

**Backend:** Node.js

**Base de datos:** Postgres



**Figura 1.** Login de la aplicación

# DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

## GUIA DE PRACTICA DE LABORATORIO /TALLER

Registrate!

Nombre \*

Apellido \*

Correo \*

Celular \*

Contraseña \*

Contraseña debe tener al menos 8 caracteres

CANCELAR GUARDAR

Figura 2. Registro de la aplicación

ADMINISTRADOR  
Eduardo Cajas

Equipos (8)

Crear Equipo

Equipos	Equipos	Equipos
<b>Analizadores de redes eléctricas 2</b> Precio: \$ 30.4 Característica: Fluke 1732 and 1734 Cantidad: 2	<b>PC</b> Precio: \$ 2.2 Característica: K Cantidad: 4	<b>Telescopio</b> Precio: \$ 20.6 Característica: aumento 2x Cantidad: 2
<b>tres</b> Precio: \$ 2 Característica: 2 Cantidad: 7	<b>Tester NL</b> Precio: \$ 20 Característica: 2 Cantidad: 2	<b>Termometro</b> Precio: \$ 20.4 Característica: Normal Cantidad: 2
<b>PC 2</b> Precio: \$ 13.5 Característica: DELL Cantidad: 2	<b>testing_1</b> Precio: \$ 2 Característica: Normal Cantidad: 2	

Logout

Figura 3. Página principal

### 4.3. Aplicación de Medidas de Seguridad:

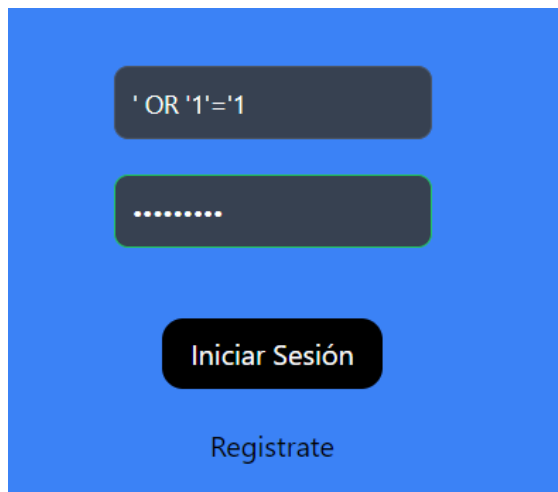
Aplicación de técnicas de prevención de ataques comunes, como inyección SQL y XSS.

#### *SQL Injection en el Login*

#### *Interceptar la Solicitud de Login:*

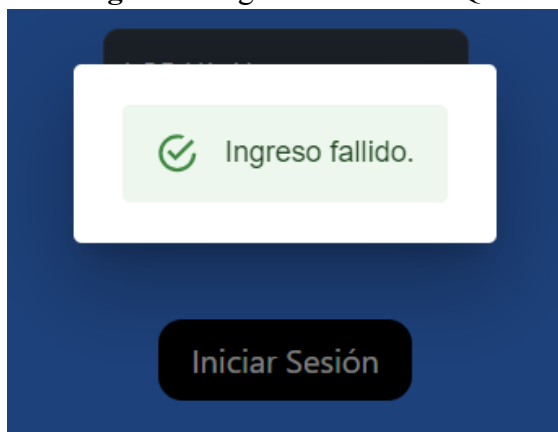
Cambia el parámetro username a algo como ' OR '1'='1'.

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**  
**GUIA DE PRACTICA DE LABORATORIO /TALLER**



A screenshot of a login interface with a blue background. At the top, there is a dark grey input field containing the SQL injection payload `' OR '1'='1`. Below it is another dark grey input field for a password, represented by dots. At the bottom, there is a black button labeled "Iniciar Sesión" and a link labeled "Registrate" below it.

**Figura 4.** Ingreso comando SQL



A screenshot of the same login interface, but with a white modal box in the center. The modal box contains a green checkmark icon and the text "Ingreso fallido." Below the modal box, the "Iniciar Sesión" button is visible.

**Figura 5.** Respuesta al ataque

**XSS**

***Campo de Usuario:***

***Intenta introducir un payload XSS en el campo de usuario, como:***

***`<script>alert('XSS')</script>`***

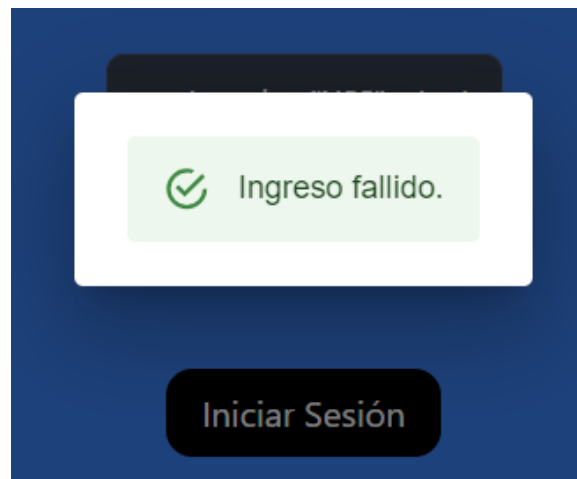


A screenshot of the login interface with a blue background. The top dark grey input field now contains the XSS payload `<script>alert('XSS')</script>`. The password field and the "Iniciar Sesión" button are also visible.

**Figura 6.** Ingreso de script

## DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

### GUIA DE PRACTICA DE LABORATORIO /TALLER



**Figura 7.** Respuesta al ataque

#### **Configuración de un firewall básico para proteger el servidor web.**

Para la configuración del firewall en el servidor web, vamos a habilitar 'ufw' (Uncomplicated Firewall) en Ubuntu Server para gestionar las reglas de acceso. Comenzamos por activar 'ufw' con 'sudo ufw enable'. A continuación, es esencial permitir solo los puertos necesarios para la operación del servidor web. Para los servicios necesarios vamos a habilitar los puertos 3000 'sudo ufw allow 3000' para el back y el puerto 5432 'sudo ufw allow 5432' para la base de datos PostgreSQL.

Esta configuración básica garantiza que solo el tráfico autorizado pueda acceder a los servicios específicos del servidor, mejorando así la seguridad general del sistema.

```
ubuntu-server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@server:/home/souser# ufw status
Status: active

To Action From
--
3000 ALLOW Anywhere
5432 ALLOW Anywhere
3000 (v6) ALLOW Anywhere (v6)
5432 (v6) ALLOW Anywhere (v6)
```

**Figura 8.** Respuesta al ataque

#### **4.4. Pruebas de Seguridad Ofensiva**

Uso de OWASP ZAP para identificar vulnerabilidades en su sistema.

# DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

## GUIA DE PRACTICA DE LABORATORIO /TALLER

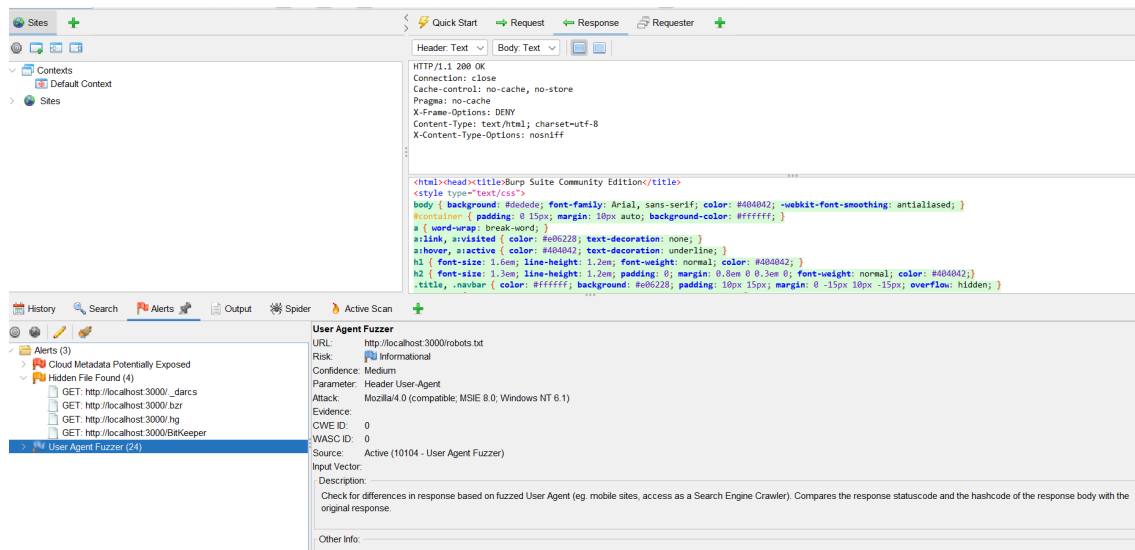


Figura 8. Análisis de vulnerabilidades

### Alert counts by alert type

This table shows the number of alerts of each alert type, together with the alert type's risk level.

(The percentages in brackets represent each count as a percentage, rounded to one decimal place, of the total number of alerts included in this report.)

Alert type	Risk	Count
<u>Cloud Metadata Potentially Exposed</u>	High	2 (66.7%)
<u>Hidden File Found</u>	Medium	8 (266.7%)
<u>User Agent Fuzzer</u>	Informational	36 (1,200.0%)
Total		3

Figura 9. Generación de reporte

### 4.5. Auditoría y Monitoreo de Seguridad

La auditoría se llevó a cabo para asegurar la seguridad de la aplicación, los cuales incluyen la revisión de actualizaciones, configuración del firewall, análisis de servicios

# **DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

## **GUIA DE PRACTICA DE LABORATORIO /TALLER**

activos, revisión de configuraciones críticas, comprobación de usuarios y permisos, revisión de logs, escaneo de vulnerabilidades.

### **Revisión de actualizaciones**

Se verificó que el sistema operativo y todos los paquetes instalados estuvieran actualizados.

### **Hallazgos**

- El sistema operativo Ubuntu 20.04 LTS está actualizado a la última versión disponible.
- Se encontraron algunos paquetes desactualizados y se aplicaron los parches de seguridad necesarios.

### **Verificación de configuración del firewall**

Se revisó la configuración del firewall 'ufw' para asegurar que solo los puertos necesarios estuvieran abiertos.

### **Hallazgos**

- El firewall se encuentra habilitado
- Los puertos abiertos son: 3000 (Aplicación Backend) y 5432 (PostgreSQL)

### **Revisión de configuraciones de servicios críticos**

Se examinó la configuración de servicios críticos como SSH, en el servidor web.

### **Hallazgos**

- SSH se encuentra deshabilitado

### **Escaneo de vulnerabilidades**

Se utilizó OWASP ZAP para realizar un escaneo de vulnerabilidades para identificar posibles debilidades.

### **Hallazgos**

- Cloud Metadata Potentially Exposed
- Hidden File Found

## **5. CONCLUSIONES:**

- La identificación de vulnerabilidades como SQL Injection y XSS no solo resalta la necesidad de un análisis técnico profundo, sino también la importancia de la educación continua y la adopción de mejores prácticas de codificación segura. La formación en



# **DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

## **GUIA DE PRACTICA DE LABORATORIO /TALLER**

seguridad informática y la implementación de estándares de codificación segura son fundamentales para minimizar los riesgos y mantener la robustez de las aplicaciones a lo largo del tiempo.

- La auditoría de seguridad realizada ha permitido identificar y corregir varias vulnerabilidades en el servidor web.

### **6. RECOMENDACIONES:**

- Utilizar funciones de escape proporcionadas por frameworks web para evitar la ejecución de scripts maliciosos.
- Auditar regularmente el código fuente y las dependencias para identificar y corregir posibles vulnerabilidades de XSS.
- Implementar y seguir las directrices establecidas en las normas de seguridad ISO/IEC 270001 y además de contemplar el TOP de vulnerabilidades de OWASP. Estas normas proporcionan un marco comprensivo para gestión de la seguridad de la información.

### **7. BIBLIOGRAFÍA**

Abikoye, O.C., Abubakar, A., Dokoro, A.H. et al. A novel technique to prevent SQL injection and cross-site scripting attacks using Knuth-Morris-Pratt string match algorithm. EURASIP J. on Info. Security 2020, 14 (2020).  
<https://doi.org/10.1186/s13635-020-00113-y>

Agarwala, D. (2021). SQL injection and XSS. International Journal of Advance Research, Ideas and Innovations in Technology. <https://www.ijariit.com>

### **8. ANEXOS**

GitHub: [https://github.com/JimeNayeli/Software\\_Seguro](https://github.com/JimeNayeli/Software_Seguro)