UNIVERSIDAD TECNOLOGICA Y POLITECNICA DE COYUCA DE BENITEZ

ASIGNATURA: FRAMEWORKS PARA DESARROLLO WEB

CUATRIMESTRE: V UNIDAD: III

PROCESO DE SEGURIDAD E IMPLEMENTACION DE LA APLICACION WEB

DOCENTE: ING. GEOVAMY PIZA RODRIGUEZ

ALUMNA: ADRIANA DE LOS SANTOS LAYNA

INGENIERIA EN TECNOLOGIAS DE LA INFORMACION E INNOVACION DIGITAL (LINEA)

20 ABRIL 2026

Indice

1. Introduccion

El presente documento constituye el reporte de la Unidad III de la asignatura Frameworks para Desarrollo Web, la cual aborda el proceso de seguridad e implementacion de aplicaciones web. A lo largo de esta unidad se trabajo con un caso practico real: el desarrollo y despliegue de una aplicacion web para la gestion de eventos universitarios de la Universidad Tecnologica y Politecnica de Coyuca de Benitez.

La aplicacion permite registrar usuarios con diferentes roles (estudiante, profesor, administrador), publicar eventos academicos como conferencias, talleres y seminarios, gestionar inscripciones y administrar todo el sistema desde un panel de administracion.

El stack tecnologico utilizado incluye HTML, CSS y JavaScript en el frontend, Node.js con Express en el backend, y MySQL como base de datos. La aplicacion fue desplegada en el dominio eventosuniversitarios.click con certificado SSL/TLS proporcionado por Let's Encrypt.

Este reporte integra las cuatro actividades de la unidad y documenta todo el proceso de seguridad, encriptacion, certificacion y despliegue implementado.

2. Desarrollo

El desarrollo completo de las 4 actividades se documenta a continuacion, abordando cada tema con detalle:

a) Tema 1 - Fundamentos de seguridad web e integracion

Concepto de seguridad web, tipos de amenazas (inyeccion SQL, XSS, CSRF, fuerza bruta, MITM, DDoS), herramientas y mecanismos de control (Helmet.js, CORS, Rate Limiting, Input Validation, Prepared Statements, CSRF Tokens, WAF).

b) Tema 2 - Mecanismos de encriptacion y control de acceso

Tipos de encriptacion (Hash/bcrypt, Simetrica/AES-256, Asimetrica/RSA, Firma digital/HMAC-SHA256). Metodos de autenticacion: basada en contrasena, basada en tokens JWT, autorizacion por roles RBAC (admin, profesor, estudiante).

c) Tema 3 - Certificados de seguridad web

Tipos de certificados SSL/TLS: DV (Domain Validation), OV (Organization Validation), EV (Extended Validation), Wildcard, Multi-Domain. Se selecciono Let's Encrypt (DV gratuito) por ser un proyecto academico. TLS 1.3 con cipher suites modernas.

d) Tema 4 - Despliegue de aplicaciones en el servidor

Servicios del servidor (Apache/Nginx, Node.js, MySQL, SSH, SFTP), permisos (644 archivos, 755 directorios, 600 .env), proceso de despliegue paso a paso, y protocolo de transferencia SFTP sobre puerto 65002.

3. Reporte Final del Caso Practico

a) Justificacion de los Mecanismos de Control de Acceso

La aplicacion implementa un sistema de control de acceso basado en roles (RBAC) con tres niveles:

* Administrador: Acceso total al sistema. Puede crear, editar, cancelar y eliminar eventos. Gestiona usuarios e inscripciones.

* Profesor: Puede crear eventos y visualizar inscripciones a sus eventos.

* Estudiante: Puede inscribirse a eventos, ver sus inscripciones y cancelarlas.

La autenticacion se implementa mediante JWT (JSON Web Tokens) firmados con HMAC-SHA256, con expiracion de 24 horas. Las contrasenas se almacenan como hashes bcrypt con salt aleatorio y 10 rondas de costo.

Adicionalmente se implementa rate limiting (5 intentos cada 15 minutos) para prevenir ataques de fuerza bruta, validacion de datos de entrada para prevenir inyeccion SQL y XSS, y tokens CSRF para proteger formularios.

b) Justificacion del Certificado de Seguridad

Se selecciono un certificado SSL/TLS DV (Domain Validation) de Let's Encrypt por las siguientes razones:

* Costo: Gratuito, adecuado para un proyecto academico.

* Nivel de validacion: DV es suficiente para una aplicacion universitaria que no procesa pagos.

* Automatizacion: Renovacion automatica cada 90 dias via Certbot/Hostinger.

* Compatibilidad: Reconocido por todos los navegadores modernos.

* Protocolo: TLS 1.3 con cipher suites modernas (AES-256-GCM, ChaCha20-Poly1305).

c) Requerimientos del Hosting

Requisito	Especificacion
Servidor web	Apache 2.4+ o Nginx 1.18+
Node.js	v18.x o superior (LTS)
MySQL	v8.0 o superior
RAM	Minimo 512 MB
Almacenamiento	Minimo 1 GB
SSL/TLS	Soporte para Let's Encrypt
Acceso SSH	Requerido para deploy y configuracion
Subdominios	Soporte para subdominios ilimitados
Proveedor utilizado	Hostinger (Plan Premium)

d) Protocolo de Transferencia de Archivos

Se selecciono SFTP (SSH File Transfer Protocol) operando sobre el puerto 65002:

* Toda la comunicacion se cifra mediante SSH.
* Las credenciales y archivos nunca viajan en texto plano.
* Se utiliza la misma autenticacion que SSH.
* Opera sobre un unico puerto (65002), simplificando reglas de firewall.
* Herramienta utilizada: FileZilla con protocolo SFTP.

Se descarto FTP tradicional por transmitir datos sin cifrado, y SCP por no soportar listado de directorios ni transferencias parciales.

e) Servicios de Autenticacion y Autorizacion

El flujo completo de autenticacion implementado:

1. El usuario accede al formulario de registro/login.
2. En registro: los datos se validan, la contrasena se hashea con bcrypt y se almacena en MySQL.
3. En login: se verifica email + contrasena hasheada. Si es valido, se genera un JWT.
4. El JWT se almacena en el cliente (localStorage) y se envia en cada request.
5. El middleware verifica el JWT y extrae el rol del usuario.
6. El middleware de autorizacion verifica si el rol tiene permiso para la accion solicitada.
7. Se registra cada acceso en logs para auditoria.

4. Conclusion

A lo largo de esta unidad se adquirieron conocimientos fundamentales sobre la seguridad e implementacion de aplicaciones web. El desarrollo del caso practico permitio aplicar de forma integral los conceptos de seguridad web, encriptacion de datos, certificados SSL/TLS y procesos de despliegue en un servidor de produccion.

La implementacion del sistema de Gestion de Eventos Universitarios demostro la importancia de considerar la seguridad como un componente transversal en todas las etapas del desarrollo: desde el diseno de la autenticacion y autorizacion, pasando por la proteccion de datos con encriptacion, hasta el despliegue seguro con HTTPS y SFTP.

El uso de tecnologias modernas como JWT para autenticacion sin estado, bcrypt para hashing de contrasenas, Let's Encrypt para certificados SSL gratuitos, y SFTP para transferencia segura de archivos, refleja las mejores practicas actuales de la industria del desarrollo web.

La aplicacion desplegada en eventosuniversitarios.click representa un ejemplo funcional de como un proyecto academico puede alcanzar estandares profesionales de seguridad y despliegue.

5. Bibliografia

- OWASP Foundation. (2021). OWASP Top 10 Web Application Security Risks.

- Mozilla Developer Network. (2026). Web Security Guidelines.

- Auth0. (2026). JWT Handbook. jwt.io.

- Let's Encrypt. (2026). Documentation. letsencrypt.org.

- Express.js. (2026). Security Best Practices. expressjs.com.

- MySQL. (2026). MySQL 8.0 Reference Manual. dev.mysql.com.

- Hostinger. (2026). Hosting Documentation. hostinger.com.