“Investigación 1: Especificación de principios de codificación segura.”

UNIVERSIDAD TECNOLÓGICA DE SAN LUIS RIO COLORADO

JAVIER VELAZQUEZ

DESARROLLO WEB INTEGRAL

Ezer Jehu Alvarado Camarillo

IDGS9-1

24 de Octubre de 2024

1. Lineamientos de la Protección de Datos Personales:

Los lineamientos para la protección de datos personales varían según las normativas de cada país o región. Sin embargo, muchos comparten los siguientes principios fundamentales:

- **Consentimiento**: Se debe obtener el consentimiento explícito del usuario para recolectar, procesar y almacenar sus datos personales.
- **Finalidad**: Los datos personales deben recopilarse para fines específicos, explícitos y legítimos, y no se deben procesar más allá de esos fines.
- **Minimización de datos**: Solo se deben recolectar los datos estrictamente necesarios para el propósito previsto.
- **Exactitud**: Los datos personales deben mantenerse exactos y actualizados.
- **Limitación de almacenamiento**: Los datos deben conservarse solo por el tiempo necesario para los fines del procesamiento.
- **Integridad y confidencialidad**: Se debe garantizar la seguridad adecuada para evitar el acceso no autorizado o la divulgación de los datos personales.
- **Transparencia**: Las personas deben estar informadas sobre cómo se usan sus datos.

Normativas clave como el **Reglamento General de Protección de Datos (GDPR)** en Europa y la **Ley de Protección de la Privacidad del Consumidor de California (CCPA)** en EE. UU. proporcionan ejemplos detallados de estos principios.

---

2. Principios de Codificación Segura:

La codificación segura implica diseñar y desarrollar software de manera que se eviten vulnerabilidades de seguridad. Algunos principios clave incluyen:

- **Validación de entradas**: Verificar y sanitizar todas las entradas de usuario para prevenir ataques de inyección (como inyección SQL o XSS).
- **Autenticación y gestión de sesiones**: Implementar controles de autenticación robustos y manejar las sesiones de usuario de manera segura.
- **Control de acceso**: Asegurarse de que los usuarios solo tengan acceso a los recursos que les corresponden según sus roles.
- **Cifrado de datos sensibles**: Encriptar datos tanto en tránsito como en reposo para proteger la confidencialidad.
- **Manejo adecuado de errores**: Evitar que los errores del sistema muestren información sensible que pueda ser aprovechada por atacantes.

- **Uso de librerías seguras**: Utilizar solo bibliotecas y componentes de terceros que sean conocidos por su seguridad y mantenerse actualizados.
- **Principio de privilegios mínimos**: Asegurarse de que el software funcione con los menores privilegios necesarios.

---

3. Puntos de Vulnerabilidad en Aplicaciones Web:

Las aplicaciones web tienen diversas áreas donde pueden ser vulnerables a ataques:

- **Inyección SQL**: Ocurre cuando las entradas no se validan adecuadamente, lo que permite a los atacantes ejecutar comandos SQL maliciosos.
- **Cross-Site Scripting (XSS)**: Inyección de scripts maliciosos en sitios web que luego son ejecutados por otros usuarios.
- **Cross-Site Request Forgery (CSRF)**: Forzando a un usuario autenticado a realizar acciones no deseadas en una aplicación en la que tiene acceso.
- **Desbordamiento de búfer**: Manipulación de la memoria de una aplicación para ejecutar código malicioso.
- **Autenticación débil**: Contraseñas simples o mecanismos de autenticación vulnerables.
- **Exposición de datos sensibles**: Datos personales o financieros expuestos debido a la falta de cifrado o mal manejo de los mismos.
- **Configuración de seguridad incorrecta**: Configuraciones predeterminadas de software que no son seguras.

---

4. Características y Diferencias entre Certificados de Seguridad SSL y TLS:

- **SSL (Secure Sockets Layer)**:
    - Protocolo de seguridad más antiguo utilizado para cifrar la comunicación entre un servidor web y el navegador.
    - Desarrollado en la década de 1990 por Netscape.
    - SSL ha sido declarado obsoleto debido a varias vulnerabilidades conocidas.
- **TLS (Transport Layer Security)**:
    - Es la evolución de SSL y se considera más seguro. TLS mejora la encriptación y la autenticación.

    - A partir de TLS 1.2 y TLS 1.3 se han introducido mejoras significativas en cuanto a rendimiento y seguridad.
    - Utiliza algoritmos más avanzados para cifrar la información en tránsito, lo que lo hace menos vulnerable a ataques.

Diferencias principales:

- **Seguridad**: TLS es más seguro que SSL, ya que soluciona vulnerabilidades conocidas de SSL.
- **Compatibilidad**: SSL ha sido reemplazado por TLS en la mayoría de los navegadores y servidores. Actualmente, cuando se habla de "SSL", en realidad se refiere a "TLS".
- **Rendimiento**: TLS es más eficiente que SSL, lo que mejora el rendimiento de las conexiones seguras.