Proyecto: Simulación y Explotación de Vulnerabilidades de Software

1. **Objetivos:**

   Desarrollar una aplicación (API y/o Frontend) que contenga vulnerabilidades de seguridad predefinidas, con el fin de simular y explotar ataques reales. El proyecto busca demostrar el impacto de fallos críticos y reforzar la concienciación en ciberseguridad mediante pruebas prácticas.

Además demostrar de manera práctica la relación entre vulnerabilidades, amenazas y los objetivos de seguridad (CID - Confidencialidad, Integridad y Disponibilidad), aplicando herramientas tecnológicas para su mitigación en un entorno controlado.

2. **Alcance:**

Implementación de Vulnerabilidades en una Aplicación (Escoger dos):

- Inyección de Comandos al Sistema Operativo (OS Command Injection): Permitir la ejecución arbitraria de comandos del SO mediante entradas no validadas.
- Inyección de Código JavaScript (XSS/Code Injection): Crear un vector de ataque para ejecutar scripts maliciosos en el lado del cliente.
- Ataque de Denegación de Servicio (DoS): Simular un colapso del sistema mediante sobrecarga de recursos.
- Hombre en el Medio (Man-in-the-Middle/MitM): Interceptar o manipular comunicaciones entre cliente y servidor.
- (Otras vulnerabilidades como SQLi, CSRF, etc.)

3. Fundamentos básicos de seguridad informática

Defina y diferencie claramente:

- Amenaza y Vulnerabilidad (con un ejemplo de cada).
- Controles de seguridad (mencione un tipo físico, lógico y administrativo).
- Gestión de riesgos (explique brevemente sus fases).

4. Evaluación del riesgo

De una forma breve clasifique y analice el riesgo de sus amenazas.

- Clasifique las amenazas/vulnerabilidades identificadas (Modelo STRIDE),
- Evalúe el riesgo de acuerdo a la probabilidad e impacto de la amenaza.

5. Simulación de Ataques:

- Simule amenazas (ej: robo de información de clientes por parte de un empleado enojado) y unas vulnerabilidades asociadas (ej: falta de validación de inputs para evitar inyecciones) en un entorno controlado.
- Explotar cada vulnerabilidad implementada, documentando el proceso paso a paso.
- Durante la simulación, identifique cómo se compromete cada objetivo de seguridad:

- Confidencialidad (ej: filtración de datos),
- Integridad (ej: manipulación de registros),
- Disponibilidad (ej: denegación de servicio).

  Explique cómo afecta la violación de un objetivo a otro (ej: ¿cómo impacta un ataque de denegación de servicio (DDoS) en la confidencialidad?).

- Utilice herramientas para simular la explotación de la vulnerabilidad
- Análisis post-simulación:

- Justifique brevemente los controles de seguridad usados para mitigar el riesgo.
- Analizar una contramedida tecnológica (herramienta, técnica o buena práctica) para mitigar la amenaza.

6. Entregables:

Código Fuente:

- Aplicación funcional con las vulnerabilidades integradas (versión final corregida).
- Scripts o herramientas utilizadas para los ataques.

Informe Ejecutivo:

- Identificación de Vulnerabilidades: Descripción técnica de cada fallo y su ubicación en el código.
- Metodología de Ataque: Pasos detallados para explotar cada vulnerabilidad, evidencias (capturas, logs).
- Recomendaciones de Mitigación: Soluciones para corregir los fallos (ej. sanitización de entradas, rate-limiting, uso de HTTPS).
- Otros aspectos solicitados en los puntos 3, 4 y 5.

7. Cronograma:

Versión Final (Miercoles, 04 de junio):

- Aplicación completa con todas las vulnerabilidades requeridas.
- Informe ejecutivo finalizado con análisis y ataques documentados.
- Versión "parcheada" con las mitigaciones aplicadas.