TEC | Tecnológico de Costa Rica

Programa del curso IC8071

Seguridad del Software

Escuela de Computación
Carrera de Ingeniería en Computación, Plan 411.

1 Datos generales

Nombre del curso:	Seguridad del Software
Código:	IC8071
Tipo de curso:	Teórico-Práctico
Electivo o no:	Si
Nº de créditos:	3
Nº horas de clase por semana:	4
Nº horas extraclase por semana:	5
Ubicación en el plan de estudios:	Curso del VI Semestre del Bachillerato de Ingeniería en Computación
Requisitos:	IC5701 Compiladores e Interpretes
Correquisitos:	Ninguno.
El curso es requisito de:	Ninguno.
Asistencia:	Obligatoria
Suficiencia:	No
Posibilidad de reconocimiento:	Sí
Vigencia del programa:	II Semestre de 2021

I parte: Aspectos relativos al plan de estudios

2 Descripción general

El curso presenta una breve reseña de los fundamentos de la seguridad del software. Donde se considerarán vulnerabilidades y vectores de ataque que pueden servir para violentar el software creado. Entre ellos: buffer overflows, SQL injection, robo de sesiones, entre otros. Además, también se considerarán defensas para prevenir o mitigar estos ataques incluyendo técnicas de programación, modelado de vulnerabilidades y algunas técnicas avanzadas de pruebas. Por otro lado, al final del curso se analizará el impacto de la *seguridad por diseño* en el proceso de desarrollo de software.

3 Objetivos

Objetivo General

Al finalizar el presente semestre, quienes atiendan el curso estarán en capacidad de diseñar soluciones de software seguro a través de mecanismos conocidos como *security by design* con el objetivo de garantizar la confidencialidad, integridad y disponibilidad de la información y las aplicaciones.

Objetivos Específicos

Al finalizar el presente semestre, quienes atiendan el curso estarán en capacidad de:

1. Conocer escenarios de vulnerabilidad y vectores de ataque en el software.
2. Evaluar las estrategias de defensa que generalmente se utilizan para prevenir y mitigar ataques hacia el software.
3. Diseñar software desde una perspectiva de seguridad sostenida a través de todo el ciclo de desarrollo de software

4 Contenidos

1. **La importancia de la Seguridad en el Software**
   - 1.1. Introducción
   - 1.2. Historia
   - 1.3. ¿Qué es seguridad del software?
   - 1.4. Complejidad, extensibilidad y conectividad
   - 1.5. Implicaciones éticas de la seguridad del software
   - 1.6. Legislación costarricense y otros referentes

2. **Patrones de Ataque**

2.1. Taxonomía de los patrones de ataque
2.2. Análisis de riesgo en el código fuente
2.3. Patrones de ataque comunes
2.4. Aplicación de los patrones de ataque

3. **Ingeniería inversa para análisis de software**
   3.1. Conceptos y herramientas
   3.2. Enfoques de ingeniería Inversa
   3.3. Métodos de reversión
   3.4. Decompilado y desensamblado
   3.5. Análisis dinámico y estático de código

4. **Explotación de software de servidores**
   4.1. Escalación de privilegios
   4.2. Encontrar puntos de inyección
   4.3. Explotación de la confianza
   4.4. Patrones de ataques específicos para software de servidores

5. **Explotación de software en clientes**
   5.1. Programas cliente como objetos de Ataque
   5.2. Cross-site Scripting (XSS)
   5.3. Client scripts y código malicioso

6. **Alteración de Entrada de Usuario**
   6.1. Detección de intrusos
   6.2. Tracing code
   6.3. Creación de solicitudes equivalentes
   6.4. Envenenamiento del proceso de auditoría (audit poisoning)

7. **Buffer overflow**
   7.1. Introducción a buffer overflow
   7.2. Vectores de inyección
   7.3. Buffer overflows en bases de datos
   7.4. Buffer overflows basado en contenido
   7.5. Stack overflow
   7.6. Heap overflow
   7.7. Payloads

8. **DevOps**
   8.1. Proceso de automatización basado en DevOps

8.2. Cifrado
8.3. Autenticación
8.4. Rootkits
8.5. Malware

9. Ciclo de vida de desarrollo de software
9.1. Buenas prácticas de seguridad de software en el ciclo de vida de desarrollo de software
9.2. Estándares de desarrollo de software seguro
9.3. Análisis de riesgo orientado a la arquitectura
9.4. Marcos de trabajo para el análisis de riesgo
9.5. Casos de abuso
9.6. Requerimientos de seguridad en software

II parte: Aspectos operativos

5 Metodología de enseñanza y aprendizaje

Se emplearán técnicas de clases magistrales y desarrollo de casos por parte del profesor, en donde se desarrollarán los aspectos teóricos y prácticos más relevantes de los diferentes temas. Además se combinarán con una alta participación por parte de los estudiantes durante el transcurso de las lecciones, por medio de llamadas orales, respuestas a casos en la pizarra y de trabajos en grupo.

Se presupone que las personas estudiantes profundizan los temas abordados en la clase en las lecturas recomendadas por el profesor y que serán responsables de desarrollar las diferentes asignaciones del curso.

El profesor asumirá el papel de facilitador y las personas estudiantes tendrán la mayor responsabilidad de su progreso.

6 Evaluación

Rubro	Porcentaje
Tareas	25%
Pruebas cortas	15%
Proyectos	40%
Examen final	20%
Total	100%

7 Bibliografía

Obligatoria

Gary McGraw. Exploting Software: How to break Code. 1 Edición. Addison Wesley

Gary MaGraw. Software Security: Building Security In. 1 Edición. Addison Wesley

Michael Howard and Steve Lipner. The Security Development Lifecycle. Microsoft Press

Adicional

Documentos publicados en el TecDigital.

8 Profesor

Dr. Herson Esquivel Vargas
h.esquivelvargas@itcr.ac.cr