

62240 - Explotación de vulnerabilidades en sistemas software

Información del Plan Docente

Año académico: 2023/24

Asignatura: 62240 - Explotación de vulnerabilidades en sistemas software

Centro académico: 110 - Escuela de Ingeniería y Arquitectura

Titulación: 534 - Máster Universitario en Ingeniería Informática

Créditos: 3.0

Curso:

Periodo de impartición: Primer semestre

Clase de asignatura: Optativa

Materia:

1. Información básica de la asignatura

El objetivo de esta asignatura es que el alumnado sea capaz de analizar código y sistemas software para la identificación y solución de las vulnerabilidades y problemas de seguridad más comunes. Así, serán capaces de aplicar diversas técnicas para analizar la seguridad y comprometer los sistemas software que sean vulnerables, demostrando fehacientemente la problemática existente y proponiendo soluciones de mejora.

Estos planteamientos y objetivos están alineados con los Objetivos de Desarrollo Sostenible (ODS) de la Agenda 2030 de Naciones Unidas (<https://www.un.org/sustainabledevelopment/es/>), en concreto, las actividades de aprendizaje previstas en esta asignatura contribuirán a al logro de las meta 9.1 del Objetivo 9 y a la meta 11.2 del Objetivo 11.

2. Resultados de aprendizaje

Cada estudiante deberá ser capaz de:

- RA1: Reconocer las vulnerabilidades más comunes en sistemas software.
- RA2: Evaluar la seguridad de un sistema software.
- RA3: Dominar diferentes técnicas de análisis de sistemas software.
- RA4: Crear pruebas de concepto que permitan comprometer la seguridad de sistemas software vulnerables.

3. Programa de la asignatura

- Introducción: Gestión de vulnerabilidades, tipos de vulnerabilidades, herramientas y laboratorio de análisis. Cuestiones éticas
- Técnicas de análisis de aplicaciones: análisis estático, análisis dinámico. Fuzzing.
- Vulnerabilidades software y técnicas de explotación: vulnerabilidades de corrupción de memoria (en heap, en pila), de enteros, de cadenas de formato, problemas de concurrencia.
- Técnicas de defensa software.
- Técnicas de explotación avanzadas. Ataques ROP, diseño de shellcodes personalizadas.

4. Actividades académicas

La dedicación del estudiante para alcanzar los resultados de aprendizaje en esta asignatura se estima en 75 horas distribuidas del siguiente modo:

- Actividades presenciales (26 horas): clases magistrales y de resolución de problemas y casos y clases prácticas de laboratorio.
- Realización de trabajos (30 horas): realización de trabajos de aplicación o investigación prácticos.
- Tutela personalizada profesor-alumno (5 horas).
- Estudio personal (10 horas).
- Evaluación (4 horas).

5. Sistema de evaluación

El estudiante deberá demostrar que ha alcanzado los resultados de aprendizaje previstos mediante las siguientes actividades de evaluación:

- Trabajos prácticos [70%]. La presentación y defensa de trabajos prácticos de programación se valorará con una calificación de prácticas que ponderará con un 70% de la nota final de la asignatura. Con esta prueba se evaluarán los resultados de aprendizaje RA1, RA2, RA4.

- Resolución de problemas [30%]. Se realizará una prueba final de evaluación, consistente en una presentación de trabajos en grupo, que servirá para demostrar que se ha logrado alcanzar los resultados de aprendizaje requeridos en la asignatura. En esta prueba se resolverán problemas de naturaleza similar a los planteados en clase (análisis de código y realización de prueba de concepto). La calificación obtenida en esta prueba ponderará un 30% de la nota final de la asignatura. Con esta prueba se evaluarán los resultados de aprendizaje RA1, RA2, RA3, y RA4.

El estudiante que no opte por el procedimiento de evaluación descrito anteriormente, o bien no supere dichas pruebas durante el periodo docente, o bien quisiera mejorar su calificación, tendrá derecho a realizar una prueba global que será programada dentro del periodo de exámenes correspondiente a la primera o segunda convocatoria.