23-24

MÁSTER UNIVERSITARIO EN INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE Y SISTEMAS INFORMÁTICOS

GUÍA DE ESTUDIO PÚBLICA

DESARROLLO DE SOFTWARE SEGURO

CÓDIGO 31105147

UNED

23-24

DESARROLLO DE SOFTWARE SEGURO

CÓDIGO 31105147

ÍNDICE

PRESENTACIÓN Y CONTEXTUALIZACIÓN
REQUISITOS Y/O RECOMENDACIONES PARA CURSAR ESTA ASIGNATURA
EQUIPO DOCENTE
HORARIO DE ATENCIÓN AL ESTUDIANTE
COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE
RESULTADOS DE APRENDIZAJE
CONTENIDOS
METODOLOGÍA
SISTEMA DE EVALUACIÓN
BIBLIOGRAFÍA BÁSICA
BIBLIOGRAFÍA COMPLEMENTARIA
RECURSOS DE APOYO Y WEBGRAFÍA
IGUALDAD DE GÉNERO

Nombre de la asignatura	DESARROLLO DE SOFTWARE SEGURO
Código	31105147
Curso académico	2023/2024
Título en que se imparte	MÁSTER UNIVERSITARIO EN INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE Y SISTEMAS INFORMÁTICOS
Tipo	CONTENIDOS
Nº ETCS	9
Horas	225
Periodo	ANUAL
Idiomas en que se imparte	CASTELLANO

PRESENTACIÓN Y CONTEXTUALIZACIÓN

Lamentablemente los denominados “ciberataques” son noticia frecuente en los medios de comunicación. Según los datos publicados por el CERT (Computer Emergency Response Team) las vulnerabilidades de los sistemas informáticos reportadas cada año crecen y aumentan su grado de sofisticación.

En este curso se presentan métodos rigurosos, técnicas y herramientas para desarrollar e implantar software seguro. Los métodos incluyen el análisis de código para detectar las vulnerabilidades habituales, la revisión de código fuente mediante herramientas de análisis estático y buenas prácticas para desarrollar código seguro en lenguajes concretos de programación.

Esta asignatura supone una extensión de los aspectos de desarrollo de software cuando se trata de desarrollar un sistema software que debe tener la cualidad adicional de ser seguro. Obviamente esta cualidad debería ser exigible en cualquier desarrollo de software actual. Sin embargo, lamentablemente los aspectos de seguridad no son tenidos en cuenta y las vulnerabilidades de los sistemas aumentan cada día más.

La asignatura Desarrollo de Software Seguro es anual, de 9 ECTS (dedicación estimada de 225 horas), de carácter optativo y perteneciente al Bloque de Ingeniería de Software. Concretamente, esta asignatura es una de las 6 que forman la materia Ingeniería del Desarrollo de Software. Las otras 5 asignaturas son: Especificación de los Sistemas Software, Arquitecturas para Sistemas Software, Arquitecturas Orientadas a Servicios, Generación Automática de Código y Desarrollo de Lineas de Producto Software mediante un enfoque generativo.

Aunque el contenido de esta asignatura puede resultar interesante para complementar el resto de materias del master, su contenido no es necesario ni recomendable para aquellos alumnos que no estén interesados en el desarrollo de un software seguro y de calidad.

REQUISITOS Y/O RECOMENDACIONES PARA CURSAR ESTA ASIGNATURA

La formación previa que deberían tener los alumnos para el adecuado seguimiento de esta asignatura son los propios de ingreso al posgrado, haciendo especial recomendación en conocimientos de ingeniería de software y lenguajes de programación.
Se recomienda que el alumno tenga preferiblemente alguna experiencia previa de programación con lenguajes C, C++, Java o similares.
El principal requisito o recomendación para cursar esta asignatura es el interés del alumno por desarrollar un software seguro y de calidad. Aunque los aspectos de seguridad evolucionan muy rapidamente, en la asignatura se abordan el origen de las vulnerabilidades partiendo de los conocimientos básicos de cualquier desarrollador. Es evidente que la experiencia en eliminar fallos anteriores resulta de gran ayuda para los nuevos ataques pero nadie puede tener todos los conocimientos previos sobre un ataque futuro.

EQUIPO DOCENTE

Nombre y Apellidos	JOSE ANTONIO CERRADA SOMOLINOS (Coordinador de asignatura)
Correo Electrónico	jcerrada@issi.uned.es
Teléfono	91398-6478
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	INGENIERÍA DE SOFTWARE Y SISTEMAS INFORMÁTICOS

Nombre y Apellidos	DAVID JOSE FERNANDEZ AMOROS
Correo Electrónico	david@issi.uned.es
Teléfono	91398-8241
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	INGENIERÍA DE SOFTWARE Y SISTEMAS INFORMÁTICOS

HORARIO DE ATENCIÓN AL ESTUDIANTE

En la metodología a distancia de la UNED, los **foros** del curso virtual son el principal recurso de atención colectiva los estudiantes. La comunicación a través de los foros tiene una doble vertiente en el aprendizaje: el enriquecimiento en el ejercicio de la dialéctica y del diálogo entre los estudiantes, por un lado, y la exposición del profesor a todos los alumnos (atención colectiva), junto con el debate que ello pueda suscitar.
En la atención colectiva de los foros del curso virtual, ante cualquier cuestión concreta, planteada sobre los contenidos o el funcionamiento de la asignatura, la respuesta será inferior a 5 días del calendario lectivo.
En cuanto a la atención individual, el equipo docente dará respuesta a través del teléfono (en el horario lectivo indicado), la solicitud de una videoconferencia (a las que se atenderán de la manera más inmediata posible) y, en horario laboral peninsular, por correo electrónico:
Profesor: *José Antonio Cerrada*
Horario de atención presencial y telefónica (*guardia*):
Jueves, lectivos, de 10:00 a 14:00

Correo electrónico: jcerrada@issi.uned.es,
Teléfono: 91 398 6478
Dirección postal:
Dpto. de Ingeniería de Software y Sistemas Informáticos. Despacho 2.23.
ETSI Informática, UNED
C/ Juan del Rosal, 16
28040 MADRID

COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE

Competencias Básicas:
CB6 - Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en el desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación
CB7 - Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio
CB8 - Que los estudiantes sean capaces de integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios
CB9 - Que los estudiantes sepan comunicar sus conclusiones y los conocimientos y razones últimas que las sustentan a públicos especializados y no especializados de un modo claro y sin ambigüedades
CB10 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

Competencias Generales:
CG01 - Saber aplicar los conocimientos adquiridos y la capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios y multidisciplinares relacionados con la Ingeniería de Sistemas y la Ingeniería de Software.
CG02 - Demostrar una comprensión sistemática del campo de estudio de la Ingeniería de Software o de la Ingeniería de Sistemas, y el dominio de las habilidades y métodos de investigación relacionados con dicho campo.
CG03 - Demostrar la capacidad de concebir, diseñar, poner en práctica y adoptar un proceso sustancial de investigación con seriedad académica.
CG04 - Ser capaz de realizar un análisis crítico, evaluación y síntesis de ideas nuevas y complejas.
CG05 - Saber comunicar sus conclusiones -y los conocimientos y razones últimas que las sustentan- a públicos especializados y no especializados, a sus colegas, a la comunidad académica en su conjunto y a la sociedad, de un modo claro y sin ambigüedades.
CG06 - Ser capaz de fomentar, en contextos académicos y profesionales, el avance tecnológico dentro de una sociedad basada en el conocimiento.
CG07 - Ser capaz de integrar conocimientos y enfrentarse a la complejidad de formular

juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios.
CG08 - Realizar una contribución a través de una investigación original que amplíe las fronteras del conocimiento desarrollando un corpus sustancial, del que parte merezca la publicación referenciada a nivel nacional o internacional.
CG09 - Poseer las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

Competencias Específicas:
CE01 - Incorporar mejoras cualitativas sustanciales, bien sea en la elaboración de software o bien en el desarrollo e implantación de sistemas robóticos.
CE02 - Concebir, implementar implantar y supervisar nuevas soluciones a los problemas específicos que se le planteen en el ámbito de la investigación, innovación y desarrollo de software o de la robótica.

RESULTADOS DE APRENDIZAJE

La asignatura está enfocada al desarrollo y mantenimiento de software seguro y sin vulnerabilidades. Por tanto, los resultados de aprendizaje que se espera que el estudiante pueda alcanzar son:

- Identificar las principales causas de vulnerabilidad conocidas y desarrollar el código seguro que las evite.
- Conocer y saber aplicar un conjunto de métodos, técnicas y herramientas que permitan probar que el software desarrollado cumple los requisitos de funcionalidad y seguridad.
- Aplicar métodos para verificar formalmente la corrección de componentes de software crítico seguro.
- Realizar, junto con las pruebas tradicionales, otras adicionales específicas de seguridad.
- Usar modelos de penetración, patrones de ataque, de abuso o mal uso del sistema en la fase de pruebas.
- Conocer los procedimientos y programas de mantenimiento de software para que continúe cumpliendo con los requisitos de funcionalidad y seguridad.

CONTENIDOS

Tema 1: Introducción

Los aspectos estudiados en este tema son los siguientes:

- Visión panorámica de las vulnerabilidades y sus costes
- Propiedades del software seguro y resiliente

Tema 2: Estudio de Vulnerabilidades

Los aspectos estudiados en este tema son los siguientes:

•Errores de programación más peligrosos según el CWE/SANS Top 25

•Conceptos de seguridad

Tema 3: Plan Estratégico

Los aspectos estudiados en este tema son los siguientes:

•Seguridad y resiliencia a lo largo del ciclo de vida

•Puntos de ataque y seguridad perimetral

•Buenas prácticas según OWASP (Open Web Application Security Project)

Tema 4: Prácticas de Desarrollo

Los aspectos estudiados en este tema son los siguientes:

•Buenas prácticas para análisis de requisitos, diseño arquitectónico y de detalle. Por ejemplo:

•Casos de uso/abuso

•Modelado de amenazas

•Análisis de riesgos

•Revisión de diseño

•Defensa en profundidad

Tema 5: Buenas Prácticas de Programación

Los aspectos estudiados en este tema son los siguientes:

•Los 10 riesgos de seguridad más críticos según OWASP

•Plataforma ESAPI (OWASP Enterprise Security API)

•Cross-Site Scripting (XSS)

•Inyección de ataques

•Autentificación y gestión de sesión

Tema 6: Gestión de Memoria en C y C++

Los aspectos estudiados en este tema son los siguientes:

•Errores más comunes de gestión de memoria

•Buffer overflow

•Stack smashing

•Validación de entradas

Tema 7: Strings, Punteros y Manejo de Enteros

Los aspectos estudiados en este tema son los siguientes:

•Errores de manejo de strings
•Errores de overflow de enteros
•Subterfugios con punteros

Tema 8: Otras vulnerabilidades en C y C++

Los aspectos estudiados en este tema son los siguientes:

•Errores de formateado de Entrada/Salida de datos
•Errores de secuenciado de Entrada/Salida de datos
•Errores de manejo de ficheros

Tema 9: Análisis Estático

Los aspectos estudiados en este tema son los siguientes:

•Tipos de análisis estático
•Herramientas de análisis
•Coverity
•Fortify

Tema 10: Pruebas

Los aspectos estudiados en este tema son los siguientes:

•Buenas prácticas de pruebas de unidades
•Pruebas de penetración
•Fuzzing

METODOLOGÍA

La docencia de esta asignatura se impartirá a distancia, siguiendo el modelo educativo propio de la UNED. El principal instrumento docente será la plataforma aLF en la que se habilitarán diversos foros para canalizar las consultas y comentarios.
Las actividades a realizar por parte del alumno se desglosan en los tres ámbitos siguientes:

•Actividades de contenido teórico: lectura de las orientaciones generales; lectura comprensiva de la bibliografía, material didáctico e información temática; e intercambio de información y consulta de dudas con el equipo docente
•Actividades de contenido práctico: manejo de herramientas informáticas y de ayuda a la presentación de resultados; intercambio de información con otros compañeros y el equipo

docente sobre aspectos prácticos y participación, argumentación y aportación constructiva en los debates en foros

•Trabajo autónomo: búsqueda de información adicional en biblioteca, Internet, etc.; selección de la información útil; actividades, que el estudiante realiza de manera autónoma, orientadas a resolver ejercicios, prácticas, problemas o trabajos que se plantean específicamente en la asignatura; realización de memorias de las prácticas, trabajos y desarrollos.

Además, el estudiante podrá realizar consultas al equipo docente a través del correo, teléfono y presencialmente en los horarios establecidos para estas actividades. Ver apartado de ***Horario de atención al estudiante*** en esta guía docente.

SISTEMA DE EVALUACIÓN

TIPO DE PRIMERA PRUEBA PRESENCIAL

Tipo de examen	No hay prueba presencial

TIPO DE SEGUNDA PRUEBA PRESENCIAL

Tipo de examen2	No hay prueba presencial

CARACTERÍSTICAS DE LA PRUEBA PRESENCIAL Y/O LOS TRABAJOS

Requiere Presencialidad	No
Descripción	

El sistema de evaluación está basado en el desarrollo de un proyecto o un conjunto de programas ad hoc en que se deben estudiar y analizar los conceptos fundamentales de la asignatura.

Para la elaboración del proyecto/programas, y su evaluación correspondiente, existen 2 posibles alternativas: por partes (Parcial) o completa previa aprobación (Completa).

En ambos casos, en el desarrollo del proyecto/programas se deben incorporar los elementos fundamentales sobre seguridad estudiados en la asignatura. Por ello, no es relevante la funcionalidad concreta del proyecto/programas pero sin embargo es fundamental estudiar el mayor número de escenarios posibles con vulnerabilidades y los correspondientes mecanismos para evitarlas.

En el caso de Evaluación Parcial es necesario presentar CINCO entregables correspondientes a cada una de las cinco partes en que se ha dividido el temario. El primer entregable se corresponderá con las tareas a realizar indicadas en el apartado de Contenidos de los temas 1 y 2. La segunda entrega se corresponderá con las tareas a realizar indicadas en los Temas 3 y 4. La tercera se corresponde con los Temas 5 y 6. La cuarta se corresponde con los Temas 7 y 8. La quinta se corresponde con los los Temas 9 y 10. Cada entregable deberá incluir una memoria explicativa, los fuentes, los ejecutables y todos aquellos elementos necesarios para poder evaluar el trabajo realizado.

Los plazos de entrega para enviar cada uno de estos entregables a través de la plataforma aLF, son los siguientes:

Entrega Uno: Desde las 0 horas del 1 de enero hasta las 23:55 del 15 de enero
Entrega Dos: Desde las 0 horas del 25 de febrero hasta las 23:55 del 1 de marzo
Entrega Tres: Desde las 0 horas del 25 de marzo hasta las 23:55 del 1 de abril
Entrega Cuatro: Desde las 0 horas del 25 de abril hasta las 23:55 del 1 de mayo
Entrega Cinco: Desde las 0 horas del 25 de mayo hasta las 23:55 del 1 de junio

Si se opta por una evaluación Completa es necesario presentar DOS entregables correspondientes a una entrega Inicial y otra Final. Las tareas a realizar se detallan en las tareas a realizar dentro del apartado de Contenidos correspondiente a todos los Temas del 1 a 10. El entregable Inicial se corresponderá con la elaboración de los requisitos funcionales y fundamentalmente de Seguridad del proyecto o programas a desarrollar, con especificaciones de uso y mal uso, riesgos y vulnerabilidades a evitar. La adecuación de la propuesta a los contenidos y objetivos de aprendizaje en la asignatura será verificada por el equipo docente y deberá ser aprobada antes de su desarrollo. En el entregable Final se deberá desarrollar la propuesta presentada y aceptada por el equipo docente a partir del entregable Inicial. Los plazos de entrega para enviar los entregables Inicial/Final a través de la plataforma aLF, son los mismos que para la evaluación Parcial. Es muy recomendable enviar el entregable Inicial lo antes posible y preferiblemente en el plazo de la Entrega Uno. En esta modalidad de evalucion Completa, los

posteriores plazos de Entrega Dos, Tres, Cuatro o Cinco servirán tanto para lograr la aceptación de la propuesta Inicial como el envío del entregable Final.
Existe la posibilidad de hacer una entrega extraordinaria en el mes de septiembre. En caso de Evaluación Completa sólo se podrá utilizar esta convocatoria si se tiene una propuesta Inicial aceptada por el equipo docente. En caso de Evaluación Parcial se utilizará para subsanar los fallos o realizar las mejoras indicadas por el equipo docente en las evaluaciones de las entregas Uno a Cinco. El plazo de entrega para enviar este entregable extraordinario a través de la plataforma aLF, es el siguiente:
Entrega Septiembre: Desde las 0 horas del 1 de septiembre hasta las 23:55 del 10 de septiembre

Criterios de evaluación

Los criterios con que se evalúan los contenidos tanto de cada entregable Parcial, como del Completo, se refieren a los objetivos de la asignatura y a los resultados del aprendizaje esperados para ella:
Identificar las principales causas de vulnerabilidad conocidas y desarrollar el código seguro que las evite.
Conocer y saber aplicar un conjunto de métodos, técnicas y herramientas que permitan probar que el software desarrollado cumple los requisitos de funcionalidad y seguridad.
Aplicar métodos para verificar formalmente la corrección de componentes de software crítico seguro.
Realizar, junto con las pruebas tradicionales, otras adicionales específicas de seguridad.
Usar modelos de penetración, patrones de ataque, de abuso o mal uso del sistema en la fase de pruebas.
Conocer los procedimientos y programas de mantenimiento de software para que continúe cumpliendo con los requisitos de funcionalidad y seguridad.
En particular, los contenidos de los entregables (bien sean los parciales o el proyecto completo) se refieren a las actividades directamente vinculadas a unos temas concretos del programa de la asignatura; por lo que los criterios de la evaluación de cada parte se refieren al cumplimiento de los objetivos de los temas que le correspondan.

Ponderación de la prueba presencial y/o los trabajos en la nota final	Segun el tipo de evaluacion elegida la ponderacion es la siguiente: Evaluacion Parcial: 20% cada uno de los 5 Entregables. Evaluacion Completa: 100%. el Entregable Final
Fecha aproximada de entrega	Las fechas estan detalladas en el apartado de Descripcion
Comentarios y observaciones	

PRUEBAS DE EVALUACIÓN CONTINUA (PEC)

¿Hay PEC? No
Descripción
Criterios de evaluación
Ponderación de la PEC en la nota final
Fecha aproximada de entrega
Comentarios y observaciones

OTRAS ACTIVIDADES EVALUABLES

¿Hay otra/s actividad/es evaluable/s? No
Descripción
Criterios de evaluación
Ponderación en la nota final
Fecha aproximada de entrega
Comentarios y observaciones

¿CÓMO SE OBTIENE LA NOTA FINAL?

La asignatura será superada cuando la nota final sea superior a cinco: NF >= 5.
La condición necesaria para obtener un resultado suficiente en la evaluación es la presentacion de las correspondientes entregas en tiempo y forma. Además, dependiendo de la modalidad de evaluación, la NF se será la siguiente:
Evaluación Parcial: NF = 0.20 x (EP1 + EP2 + EP3 + EP4 + EP5) donde EP es el Entregable Parcial correspondiente.
Evaluación Completa: NF = EF, donde EF es el Entregable Final.

BIBLIOGRAFÍA BÁSICA

ISBN(13): 9780321822130
Título: SECURE CODING IN C AND C++ Second Edition edición
Autor/es: Robert C. Seacord
Editorial: ADDISON WESLEY

ISBN(13): 9781439826966
Título: SECURE AND RESILENT SOFTWARE DEVELOPMENT
Autor/es: Mark S. Merkow And Lakshmikanth Raghavan
Editorial: CRC Press

Los dos libros están accesibles desde el portal de la UNED. Hay que autenticarse, y a partir de ahí.

•El libro de Seacord está aquí: (enlace)

•El libro de Merkow está aquí: (enlace)

Hay un artículo que forma parte de la biblografía recomendada:

•Tsipenyuk, Katrina; Chess, Brian &McGraw, Gary. ***Seven Pernicious Kingdoms: A Taxonomy of Software Security Errors.*** IEEE Security &Privacy, 2005

El artículo está disponible en el curso virtual de la asignatura.

BIBLIOGRAFÍA COMPLEMENTARIA

Aunque no se consideran necesarios para el estudio de la asignatura, los libros y documentos de esta bibliografía complementaria pueden ser muy interesantes para un estudio en mayor profundidad de la asignatura.

La relación de documentos es la siguiente:

•The MITRE Corporation (MITRE). ***Common Weakness Enumeration. (2010)*** http://cwe.mitre.org/

•Grembi, Jason. ***Secure Software Development - A Security Programmer's Guide.*** Tutorial at 11th Semi-Annual Software Assurance Forum. Arlington, VA, November 2009. Software Engineering Institute, Carnegie Mellon University, 2009.
https://www.vte.cert.org/vteweb/go/2699.aspx

•Gerhart, Susan; Hogle, Jan; &Crandall, Jedidiah. ***How Do Buffer Overflow Attacks Work? (2002).***
http://nsfsecurity.pr.erau.edu/bom/
En este documento se incluye una introducción a los problemas de buffer overflows y stack-smashing con ejercicios y animaciones muy interesantes que el alumno puede y debe utilizar para el estudio de la asignatura.

•CERT. ***CERT Secure Coding Standards (2010).***
https://www.securecoding.cert.org/

•Miller, Barton P.; Cooksey, Gregory; &Moore, Fredrick. ***An Empirical Study of the Robustness of MacOS Applications Using Random Testing.*** ACM SIGOPS Operating Systems Review 41, 1 (January 2007): 78-86.

•Golze, Andreas; Sarbiewski, Mark; &Zahm, Alain. ***Optimize Quality for Business Outcomes: A Practical Approach to Software Testing.*** Wiley Publishing, 2008. El capítulo 8 sobre pruebas de seguridad es especialmente útil.

•Howard, Michael &LeBlanc, David. ***Writing Secure Code, 2nd ed.*** Microsoft Press, 2003.

•Howard, Michael; LeBlanc, David; &Viega, John. ***19 Deadly Sins of Software Security.*** McGraw-Hill, 2005.

RECURSOS DE APOYO Y WEBGRAFÍA

En la plataforma aLF se incluye todo aquel material (articulos, cronograma de actividades, etc.) que resulte interesante para el alumno. Asimismo, se recuerda que los libros básicos para el estudio de la asignatura están disponibles en Safari books mediante la subscripción a la que todos los alumnos de la UNED tienen acceso. No hace falta comprarlos. Se puede acceder pinchando en safari books online a través del siguiente link: http://portal.uned.es/portal/page?_pageid=93,26012339&_dad=portal&_schema=PORTAL

IGUALDAD DE GÉNERO

En coherencia con el valor asumido de la igualdad de género, todas las denominaciones que en esta Guía hacen referencia a órganos de gobierno unipersonales, de representación, o miembros de la comunidad universitaria y se efectúan en género masculino, cuando no se hayan sustituido por términos genéricos, se entenderán hechas indistintamente en género femenino o masculino, según el sexo del titular que los desempeñe.