Programa de la asignatura Seguridad Informática

presentado como parte de los requisitos para el Concurso Interno FCEIA N 969

Profesor Asociado dedicación Semiexclusiva

Maximiliano Cristiá

1. Introducción

Seguridad Informática es una asignatura del primer semestre de la Licenciatura en Ciencias de la Computación (LCC) de la Facultad de Ciencias Exactas, Ingeniería y Agrimensura (UNR). El presente programa se presenta como parte de los requisitos solicitados para el concurso que provee un cargo de Profesor Asociado dedicación Semiexclusiva.

Según el Plan de Estudios (2010) de LCC, Seguridad Informática pertenece al área de Ingeniería de Software, Bases de Datos y Sistemas de información. La asignatura debe dictarse en 90 horas[1], a razón de 6 horas semanales, de las cuales 45 corresponden a práctica y laboratorio. Por otra parte solo Ingeniería de Software I es correlativa con Seguridad Informática. Según la misma fuente la delimitación de contenidos de la asignatura es la siguiente:

> Sistemas de Información. Teoría general de Sistemas. Privacidad, integridad y seguridad en sistemas de información. Nociones de Auditoría y Peritaje. Protocolos de encriptación y autenticación: Kerberos, Leighton-Micali, pruebas de conocimiento cero, MD5, etc.

2. Fundamentación de la materia dentro del plan de estudios y en relación al perfil del graduado

Según el plan de estudios 2010 de LCC, dentro de las incumbencias del título de LCC (Resolución 786/2009 de fecha 26/05/2009) se encuentran las siguientes relacionadas con la seguridad de la información[2]:

2. Establecer métricas y normas de calidad y seguridad de software, controlando las mismas a fin de tener un producto industrial software que respete las normas nacionales e internacionales. Estas normas definen los procesos de especificación formal del producto, de control del diseño, desarrollo, implementación y mantenimiento. Definición de métricas de validación y certificación de calidad.
4. Efectuar las tareas de Auditoría de los Sistemas Informáticos. Realizar arbitrajes, pericias y tasaciones relacionados con los Sistemas Informáticos.

[1] Asumiendo un cuatrimestre de 15 semanas de clase

[2] Transcrito textualmente de la fuente mencionada.

7. Planificar, dirigir, realizar y/o evaluar los sistemas de seguridad en el almacenamiento y procesamiento de la información. Especificación, diseño, implementación y mantenimiento de los componentes de seguridad de información en los sistemas de software de aplicación. Establecimiento y control de metodologías de procesamiento de datos orientadas a seguridad incluyendo las de datawarehousing.

Por otra parte el plan de estudios en cuestión menciona en varios puntos la necesidad de que los egresados de LCC sean capaces de trabajar con especificaciones formales (ítem 1, 2 y 8); y establece que un egresado de LCC podrá realizar tareas de investigación científica básica y aplicada en Informática.

En consecuencia, el programa que se presenta contempla aspectos tanto teóricos como prácticos con el fin de que los egresados puedan iniciar su actividad profesional en la industria o comenzar una carrera académica. El programa pone atención en cuestiones de Seguridad Informática relacionadas con la formalización de propiedades de seguridad de los sistemas de software.

3. Objetivos de la materia

Al finalizar la asignatura los alumnos deberán:

- Conocer el vocabulario y los conceptos básicos de la Seguridad Informática.
- Comprender la complejidad del problema de la confidencialidad en sistemas de cómputo abiertos y conocer las principales soluciones propuestas.
- Comprender el alcance de la seguridad basada en lenguajes de programación.
- Conocer los errores de programación más habituales que pueden dar lugar a vulnerabilidades.
- Ser capaces de colaborar en el desarrollo de software que deba cumplir con requisitos de seguridad.
- Comprender el alcance de la criptografía como herramienta de la Seguridad Informática.
- Manejar los fundamentos de la criptografía y conocer con cierta profundidad algunos de los métodos y protocolos criptográficos clásicos.
- Conocer al menos un método de análisis de protocolos criptográficos.
- Conocer algunas normas de Seguridad Informática con el fin de poder colaborar en tareas de auditoría.
- Conocer cuestiones básicas de otros problemas de la Seguridad Informática tales como: autenticación, autorización, auditoría, integridad, privacidad, disponibilidad; etc.

4. Programa analítico

Unidad I Vocabulario y Conceptos Básicos

I.1. Introducción histórica a la seguridad informática

I.2. Problemas de la seguridad informática

I.3. Conceptos básicos: definición de seguridad informática; confidencialidad, integridad y disponibilidad; principios de seguridad informática; vulnerabilidad, adversario y riesgo; política de seguridad; identificación, autenticación, autorización y auditoría; control de acceso; seguridad en las comunicaciones; *security* y *safety*; privacidad; etc.

Unidad II Confidencialidad en Sistemas de Cómputo

- II.1. El problema de la confidencialidad en sistemas de cómputo
- II.2. Control de acceso discrecional
- II.3. Control de acceso obligatorio; seguridad multi-nivel
- II.4. Flujo de información
- II.5. Nointerferencia
- II.6. Multi-ejecución nointerferente
- II.7. Otras aproximaciones al problema

Unidad III Seguridad en Lenguajes de Programación

- III.1. Evitar o detectar errores de seguridad en la implementación
- III.2. Utilización de características de los lenguajes para implementar políticas de seguridad
- III.3. Ejemplos de errores de programación que producen fallas de seguridad: desbordamiento de arreglos e inyección de sentencias SQL
- III.4. Prácticas de programación segura
- III.5. Introducción a algunos tópicos avanzados
  - III.5.1. Código que incluye pruebas de corrección (*proof-carrying code*)
  - III.5.2. Monitores de referencia en línea
  - III.5.3. Ofuscación de código

Unidad IV Introducción a la Criptografía

- IV.1. Conceptos básicos de criptografía: relación entre criptografía y seguridad informática; algoritmo criptográfico; claves de encriptación; texto legible y encriptado; etc.
- IV.2. Criptografía de clave simétrica: introducción al algoritmo DES
- IV.3. Criptografía de clave asimétrica: introducción al algoritmo RSA
- IV.4. Resúmenes de mensajes y firmas electrónicas

Unidad V Introducción al Análisis de Protocolos Criptográficos

- V.1. Protocolos criptográficos: aplicaciones y ejemplos.
- V.2. Introducción a la lógica de Burrows, Abadi y Needham (BAN)

Unidad VI Introducción a la Auditoría de Software

- VI.1. Norma ISO 27001
- VI.2. Normativa de la industria de pagos con tarjeta para auditoría de sistemas de pagos electrónicos
- VI.3. Introducción a las Comunicaciones A 4192 y A 4609 del Banco Central de la República Argentina

5. Bibliografía de la materia

A continuación se detalla la bibliografía para cada unidad del programa (ordenada alfabéticamente por autor). Algunas de las referencias son deliberadamente "antiguas" con el fin de darles a los alumnos "la punta del ovillo" a partir de la cual realizar algunos trabajos prácticos (ver más adelante).

Unidad I

[1] Marshall D. Abrams, Sushil G. Jajodia, and H. J. Podell, editors. *Information Security: An Integrated Collection of Essays*. IEEE Computer Society Press, Los Alamitos, CA, USA, 1st edition, 1995.

[2] M. Bishop. *Computer Security: Art and Science*. Addison-Wesley, 2003.

[3] Morrie Gasser. *Building a Secure Computer System*. Van Nostrand Reinhold Co., New York, NY, USA, 1988.

Unidad II

[1] Marshall D. Abrams, Sushil G. Jajodia, and H. J. Podell, editors. *Information Security: An Integrated Collection of Essays*. IEEE Computer Society Press, Los Alamitos, CA, USA, 1st edition, 1995.

[2] D. Elliot Bell and Leonard LaPadula. Secure computer systems: Mathematical foundations. MTR 2547, The MITRE Corporation, May 1973.

[3] D. Elliot Bell and Leonard LaPadula. Secure computer systems: Mathematical model. ESD-TR 73-278, The MITRE Corporation, November 1973.

[4] Maximiliano Cristiá and Pablo Mata. Runtime enforcement of noninterference by duplicating processes and their memories. In *Workshop de Seguridad Informática WSEGI 2009*, Mar del Plata, Argentina, 2009. SADIO.

[5] Dorothy E. Denning. A lattice model of secure information flow. *Commun. ACM*, 19(5):236–243, May 1976.

[6] Morrie Gasser. *Building a Secure Computer System*. Van Nostrand Reinhold Co., New York, NY, USA, 1988.

[7] Joseph A. Goguen and José Meseguer. Security policies and security models. In *IEEE Symposium on Security and Privacy*, pages 11–20, 1982.

[8] A. Sabelfeld and A. C. Myers. Language-based information-flow security. *IEEE J.Sel. A. Commun.*, 21(1):5–19, September 2006.

Unidad III

[1] Aleph One. Smashing the stack for fun and profit.

[2] M. Bishop. *Computer Security: Art and Science*. Addison-Wesley, 2003.

[3] Úlfar Erlingsson and Fred B. Schneider. SASI enforcement of security policies: A retrospective. In *Proceedings of the 1999 Workshop on New Security Paradigms*, NSPW '99, pages 87–95, New York, NY, USA, 2000. ACM.

[4] James H. Morris, Jr. Protection in programming languages. *Commun. ACM*, 16(1):15–21, January 1973.

[5] George C. Necula. Proof-carrying code. In *Proceedings of the 24th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '97, pages 106–119, New York, NY, USA, 1997. ACM.

[6] OWASP Foundation. *OWASP Code Review*. OWASP Foundation, 2009.

[7] Ta-chung Tsai, Alejandro Russo, and John Hughes. A library for secure multi-threaded information flow in haskell. In *Proceedings of the 20th IEEE Computer Security Foundations Symposium*, CSF '07, pages 187–202, Washington, DC, USA, 2007. IEEE Computer Society.

Unidad IV

[1] Niels Ferguson and Bruce Schneier. *Practical cryptography*. Wiley, 2003.

[2] Niels Ferguson, Bruce Schneier, and Tadayoshi Kohno. *Cryptography Engineering - Design Principles and Practical Applications*. Wiley, 2010.

[3] Bruce Schneier. *Applied cryptography - protocols, algorithms, and source code in C (2. ed.)*. Wiley, 1996.

Unidad V

[1] M. Bishop. *Computer Security: Art and Science*. Addison-Wesley, 2003.

[2] Michael Burrows, Martin Abadi, and Roger Needham. A logic of authentication. *ACM Trans. Comput. Syst.*, 8(1):18–36, February 1990.

Unidad VI

[1] Argencard S.A. Payment Card Security Industry: Procedimientos de Auditoría.

[2] BCRA. Comunicación A 4192

[3] BCRA. Comunicación A 4609

[4] ISO. Texto del estándar ISO 27001.

6. Metodología de enseñanza y aprendizaje

El postulante lleva 17 años a cargo de Ingeniería de Software I y II y dictó durante 7 años Informática I (Escuela de Formación Básica – FCEIA). Esta experiencia le ha demostrado que una separación taxativa entre teoría y práctica no resulta la mejor estrategia pedagógica para que los alumnos aprendan los fundamentos del desarrollo de software, y en particular de la Seguridad Informática. Se considera que el responsable de evitar esa separación es el docente a cargo del dictado de las clases de "teoría". Por lo tanto se propone que las clases de teoría estén vertebradas por un problema práctico que requiera la introducción de nuevos fundamentos teóricos. Es decir, presentar la teoría como fundamento de la práctica y no como algo de lo cual se deriva una cierta forma de resolver problemas. Por ejemplo, el concepto de noninteferencia surge a partir de la necesidad de resolver un problema práctico (preservar un secreto almacenado en una computadora que es utilizada por más de un usuario); y no como un concepto elaborado por un genio que mágicamente sirve para solucionar ciertos problemas, de los cuales el alumno desconoce hasta su existencia.

En cuanto a la evaluación de los alumnos se propone realizar un trabajo práctico, relativamente sencillo y corto, por cada unidad del programa con el fin de fijar los conocimientos adquiridos. Cada trabajo práctico podrá ser realizado individualmente o en grupos de hasta 3 personas dependiendo del trabajo. Cada trabajo práctico será definido junto a los docentes que tomen a su cargo la práctica de la materia. La regularidad o promoción de la materia se alcanzará al entregar y aprobar todos los trabajos prácticos dentro de los plazos establecidos para cada caso. La aprobación final de la materia se hará al entregar un trabajo práctico final a elección del alumno. Los únicos requisitos para este trabajo serán:

1. Deberá ser esencialmente sobre un tema de seguridad informática
2. Deberá ser algo de elaboración própia
3. Deberá ser algo que vaya un poco más allá de lo visto durante el cursado

Es decir la cátedra no fijará temas a priori los cuales podrán ser elegidos libremente por los estudiantes. Ejemplos de posibles trabajos finales son: un estudio del estado del arte razonable sobre alguno de los temas vistos en clase o sobre otros (por ejemplo, comenzando con alguno de los artículos utilizados en clase encontrar y estudiar las, digamos, cinco siguientes referencias importantes; esta es la razón por la cual se propone utilizar las referencias fundacionales de ciertos temas); implementación de algunas de las ideas vistas en clase u otras cuestiones de seguridad informática; analizar detalladamente una o más vulnerabilidades; encontrar vulnerabilidades; explotar vulnerabilidades; etc.