Universidad Católica San Pablo (UCSP)
Escuela Profesional de
Ciencia de la Computación
SILABO

CS3I1. Seguridad en Computación (Obligatorio)

2021-II

1. Información general		
1.1 Escuela	:	Ciencia de la Computación
1.2 Curso	:	CS3I1. Seguridad en Computación
1.3 Semestre	:	8^{vo} Semestre.
1.4 Prerrequisitos	:	CS231. Redes y Comunicación. (7^{mo} Sem)
1.5 Condición	:	Obligatorio
1.6 Modalidad de aprendizaje	:	Virtual
1.7 horas	:	1 HT; 2 HP; 2 HL;
1.8 Créditos	:	3

2. Profesores

Titular

- Julio Omar Santisteban Pablo <jsantisteban@ucsp.edu.pe>
  - Doctor en Ciencias de la Computación, Universidad Nacional de San Agustin, Perú, 2021.
  - Master en Internetworking, University of Technology, Australia, 2008.

3. Fundamentación del curso

Hoy en dia la información es uno de los activos más preciados en cualquier organización. Este cursos está orientado a poder brindar al alumno los elementos de seguridad orientados a proteger la información de la organización y principalmente poder preveer los posibles problemas relacionados con este rubro. Esta materia involucra el desarrollo de una actitud preventiva por parte del alumno en todas las áreas relacionadas al desarrollo de software.

4. Resumen

1. Fundamentos y Conceptos en Seguridad 2. Principios de Diseño Seguro 3. Programación Defensiva 4. Ataques y Amenazas 5. Seguridad de Red 6. Criptografía 7. Seguridad en la Web 8. Seguridad de plataformas 9. Investigación digital (Digital Forensics) 10. Seguridad en Ingeniería de Software

5. Objetivos Generales

- Discutir a un nivel intermedio avanzado los los fundamentos de la Seguridad Informática.
- Brindar los diferentes aspectos que presenta el código malicioso.
- Que el alumno conozca los conceptos de criptografía y seguridad en redes de computadoras.
- Discutir y analizar junto con el alumno los aspectos de la Seguridad en Internet.

6. Contribución a los resultados (*Outcomes*)

Esta disciplina contribuye al logro de los siguientes resultados de la carrera:

a) Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (**Usar**)

b) Analizar problemas e identificar y definir los requerimientos computacionales apropiados para su solución. (**Evaluar**)

c) Diseñar, implementar y evaluar un sistema, proceso, componente o programa computacional para alcanzar las necesidades deseadas. (**Evaluar**)

g) Analizar el impacto local y global de la computación sobre los individuos, organizaciones y sociedad. (**Evaluar**)

h) Incorporarse a un proceso de aprendizaje profesional continuo. (**Usar**)

i) Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. (**Evaluar**)

j) Aplicar la base matemática, principios de algoritmos y la teoría de la CS en el modelamiento y diseño de sistemas. (**Usar**)

7. Contenido

UNIDAD 1: Fundamentos y Conceptos en Seguridad (25)

Competencias: a,g

Contenido	**Objetivos Generales**
• CIA (Confidencialidad, Integridad, Disponibilidad) • Conceptos de riesgo, amenazas, vulnerabilidades, y los tipos de ataque . • Autenticación y autorización, control de acceso (vs. obligatoria discrecional) • Concepto de la confianza y la honradez . • Ética (revelación responsable)	• Analizar las ventajas y desventajas de equilibrar las propiedades clave de seguridad(Confidenciabilidad, Integridad, Disponibilidad) [Familiarizarse] • Describir los conceptos de riesgo, amenazas, vulnerabilidades y vectores de ataque(incluyendo el hecho de que no existe tal cosa como la seguridad perfecta) [Familiarizarse] • Explicar los conceptos de autentificación, autorización, control de acceso [Familiarizarse] • Explicar el concepto de confianza y confiabilidad [Familiarizarse] • Reconocer de que hay problemas éticos más importantes que considerar en seguridad computacional, incluyendo problemas éticos asociados a arreglar o no arreglar vulnerabilidades y revelar o no revelar vulnerabilidades [Familiarizarse]

Lecturas: W and L (2014)

UNIDAD 2: Principios de Diseño Seguro (25)	
Competencias: a,g,h	
Contenido	**Objetivos Generales**
• Menor privilegio y aislamiento. • Valores predeterminados a prueba de fallos. • Diseño abierto. • La seguridad de extremo a extremo. • La defensa en profundidad (por ejemplo, la programación defensiva, defensa en capas) • Diseño de seguridad. • Las tensiones entre la seguridad y otros objetivos de diseño. • Mediación completa. • El uso de componentes de seguridad vetados. • Economía del mecanismo (la reducción de la base informática de confianza, minimizar la superficie de ataque) • Seguridad utilizable. • Componibilidad de seguridad. • Prevención, detección y disuasión.	• Describir el principio de privilegios mínimos y el aislamiento que se aplican al diseño del sistema [Familiarizarse] • Resumir el principio de prueba de fallos y negar por defecto [Familiarizarse] • Discutir las implicaciones de depender de diseño abierto o secreto de diseño para la seguridad [Familiarizarse] • Explicar los objetivos de seguridad de datos de extremo a extremo [Familiarizarse] • Discutir los beneficios de tener múltiples capas de defensas [Familiarizarse] • Por cada etapa en el ciclo de vida de un producto, describir que consideraciones de seguridad deberian ser evaluadas [Familiarizarse] • Describir el costo y ventajas y desventajas asociadas con el diseño de seguridad de un producto. [Familiarizarse] • Describir el concepto de mediación y el principio de mediación completa [Familiarizarse] • Conocer los componentes estándar para las operaciones de seguridad, en lugar de reinventar las operaciones fundamentales [Familiarizarse] • Explicar el concepto de computación confiable incluyendo base informática confiable y de la superficie de ataque y el principio de minimización de base informática confiable [Familiarizarse] • Discutir la importancia de la usabilidad en el diseño de mecanismos de seguridad [Familiarizarse] • Describir problemas de seguridad que surgen en los límites entre varios componentes [Familiarizarse] • Identificar los diferentes roles de mecanismos de prevención y mecanismos de eliminación/disuación [Familiarizarse]
Lecturas: W and L (2014)	

UNIDAD 3: Programación Defensiva (25)

Competencias: b,i

Contenido	Objetivos Generales

• Validación de datos de entrada y sanitización
• Elección del lenguaje de programación y lenguajes con tipos de datos seguro.
• Ejemplos de validación de entrada de datos y sanitización de errores.
  – Desbordamiento de búfer
  – Errores enteros
  – Inyección SQL
  – Vulnerabilidad XSS
• Las condiciones de carrera.
• Manejo correcto de las excepciones y comportamientos inesperados.
• Uso correcto de los componentes de terceros.
• Desplegar eficazmente las actualizaciones de seguridad.
• Información de control de flujo.
• Generando correctamente el azar con fines de seguridad.
• Mecanismos para la detección y mitigación de datos de entrada y errores de sanitización.
• Fuzzing
• El análisis estático y análisis dinámico.
• Programa de verificación.
• Soporte del sistema operativo (por ejemplo, la asignación al azar del espacio de direcciones, canarios)
• El soporte de hardware (por ejemplo, el DEP, TPM)
	• Explicar por que la validación de entrada y desinfección de datos es necesario en el frente del control contencioso del canal de entrada [Usar]
	• Explicar por que uno deberia escoger para desallorrar un programa en un lenguaje tipo seguro como Java, en contraste con un lenguaje de programación no seguro como C/C++ [Usar]
	• Clasificar los errores de validación de entrada común, y escribir correctamente el código de validación de entrada [Usar]
	• Demostrar el uso de un lenguaje de programación de alto nivel cómo prevenir una condición de competencia que ocurran y cómo manejar una excepción [Usar]
	• Demostrar la identificación y el manejo elegante de las condiciones de error [Familiarizarse]
	• Explique los riesgos de mal uso de las interfaces con código de terceros y cómo utilizar correctamente el código de terceros [Familiarizarse]
	• Discutir la necesidad de actualizar el software para corregir las vulnerabilidades de seguridad y la gestión del ciclo de vida de la corrección [Familiarizarse]

Lecturas: W and L (2014)

UNIDAD 4: Ataques y Amenazas (25)	
Competencias: b,i	
Contenido	**Objetivos Generales**
• Atacante metas, capacidades y motivaciones (como economía sumergida, el espionaje digital, la guerra cibernética, las amenazas internas, hacktivismo, las amenazas persistentes avanzadas) • Los ejemplos de malware (por ejemplo, virus, gusanos, spyware, botnets, troyanos o rootkits) • Denegación de Servicio (DoS) y Denegación de Servicio Distribuida (DDoS) • Ingeniería social (por ejemplo, perscando) • Los ataques a la privacidad y el anonimato . • El malware / comunicaciones no deseadas, tales como canales encubiertos y esteganografía.	• Describir tipos de ataques similares en contra de un sistema en particular [Familiarizarse] • Discutir los limitantes de las medidas en contra del malware (ejm. detección basada en firmas, detección de comportamiento) [Familiarizarse] • Identificar las instancias de los ataques de ingeniería social y de los ataques de negación de servicios [Familiarizarse] • Discutir como los ataques de negación de servicos puede ser identificados y reducido [Familiarizarse] • Describir los riesgos de la privacidad y del anonimato en aplicaciones comunmente usadas [Familiarizarse] • Discutir los conceptos de conversión de canales y otros procedimientos de filtrado de datos [Familiarizarse]
Lecturas: W and L (2014)	

UNIDAD 5: Seguridad de Red (25)	
Competencias: b,i	
Contenido	**Objetivos Generales**
• Red de amenazas y tipos de ataques específicos (por ejemplo, la denegación de servicio, spoofing, olfateando y la redirección del tráfico, el hombre en el medio, ataques integridad de los mensajes, los ataques de enrutamiento, y el análisis de tráfico) • El uso de cifrado de datos y seguridad de la red . • Arquitecturas para redes seguras (por ejemplo, los canales seguros, los protocolos de enrutamiento seguro, DNS seguro, VPN, protocolos de comunicación anónimos, aislamiento) • Los mecanismos de defensa y contramedidas (por ejemplo, monitoreo de red, detección de intrusos, firewalls, suplantación de identidad y protección DoS, honeypots, seguimientos) • Seguridad para redes inalámbricas, celulares . • Otras redes no cableadas (por ejemplo, ad hoc, sensor, y redes vehiculares) • Resistencia a la censura. • Gestión de la seguridad operativa de la red (por ejemplo, control de acceso a la red configure)	• Describir las diferentes categorías de amenazas y ataques en redes [Familiarizarse] • Describir las arquitecturas de criptografía de clave pública y privada y cómo las ICP brindan apoyo a la seguridad en redes [Familiarizarse] • Describir ventajas y limitaciones de las tecnologías de seguridad en cada capa de una torre de red [Familiarizarse] • Identificar los adecuados mecanismos de defensa y sus limitaciones dada una amenaza de red [Usar]
Lecturas: W and L (2014)	

UNIDAD 6: Criptografía (25)

Competencias: b,i

Contenido	Objetivos Generales
• Terminología básica de criptografía cubriendo las nociones relacionadas con los diferentes socios (comunicación), canal seguro / inseguro, los atacantes y sus capacidades, cifrado, descifrado, llaves y sus características, firmas. • Tipos de cifrado (por ejemplo, cifrado César, cifrado affine), junto con los métodos de ataque típicas como el análisis de frecuencia. • Apoyo a la infraestructura de clave pública para la firma digital y el cifrado y sus desafíos. • Criptografía de clave simétrica: – El secreto perfecto y el cojín de una sola vez – Modos de funcionamiento para la seguridad semántica y encriptación autenticada (por ejemplo, cifrar-entonces-MAC, OCB, GCM) – Integridad de los mensajes (por ejemplo, CMAC, HMAC) • La criptografía de clave pública: – Permutación de trampilla, por ejemplo, RSA – Cifrado de clave pública, por ejemplo, el cifrado RSA, cifrado El Gamal – Las firmas digitales – Infraestructura de clave pública (PKI) y certificados – Supuestos de dureza, por ejemplo, Diffie-Hellman, factoring entero • Protocolos de intercambio de claves autenticadas, por ejemplo, TLS . • Primitivas criptográficas: – generadores pseudo-aleatorios y cifrados de flujo – cifrados de bloque (permutaciones pseudo-aleatorios), por ejemplo, AES – funciones de pseudo-aleatorios – funciones de hash, por ejemplo, SHA2, resistencia colisión – códigos de autenticación de mensaje – funciones derivaciones clave	• Describir el propósito de la Criptografía y listar formas en las cuales es usada en comunicación de datos [Familiarizarse] • Definir los siguientes términos: Cifrado, Criptoanálisis, Algorítmo Criptográfico, y Criptología y describe dos métodos básicos (cifrados) para transformar texto plano en un texto cifrado [Familiarizarse] • Discutir la importancia de los números primos en criptografía y explicar su uso en algoritmos criptográficos [Familiarizarse] • Ilustrar como medir la entropía y como generar aleatoriedad criptográfica [Usar] • Usa primitivas de clave pública y sus aplicaciones [Usar] • Explicar como los protocolos de intercambio de claves trabajan y como es que pueden fallar [Familiarizarse] • Discutir protocolos criptográficos y sus propiedades [Familiarizarse]

Lecturas: W and L (2014)

UNIDAD 7: Seguridad en la Web (25)

Competencias: a,g

Contenido	**Objetivos Generales**
• Modelo de seguridad Web – Modelo de seguridad del navegador incluida la política de mismo origen – Los límites de confianza de cliente-servidor, por ejemplo, no pueden depender de la ejecución segura en el cliente • Gestión de sesiones, la autenticación: – Single Sign-On – HTTPS y certificados • Vulnerabilidades de las aplicaciones y defensas : – Inyección SQL – XSS – CSRF • Seguridad del lado del cliente : – Política de seguridad Cookies – Extensiones de seguridad HTTP, por ejemplo HSTS – Plugins, extensiones y aplicaciones web – Seguimiento de los usuarios Web • Herramientas de seguridad del lado del servidor, por ejemplo, los cortafuegos de aplicación Web (WAFS) y fuzzers	• Describe el modelo de seguridad de los navegadores incluyendo las políticas del mismo origen y modelos de amenazas en seguridad web [Familiarizarse] • Discutir los conceptos de sesiones web, canales de comunicación seguros tales como Seguridad en la Capa de Transporte(TLS) y la importancia de certificados de seguridad, autenticación incluyendo inicio de sesión único, como OAuth y Lenguaje de Marcado para Confirmaciones de Seguridad($SAML$) [Familiarizarse] • Investigar los tipos comunes de vulnerabilidades y ataques en las aplicaciones web, y defensas contra ellos [Familiarizarse] • Utilice las funciones de seguridad del lado del cliente [Usar]

Lecturas: W and L (2014)

UNIDAD 8: Seguridad de plataformas (25)	
Competencias: b,i	
Contenido	**Objetivos Generales**
• Integridad de código y firma de código. • Arranque seguro, arranque medido, y la raíz de confianza. • Testimonio. • TPM y coprocesadores seguros. • Las amenazas de seguridad de los periféricos, por ejemplo, DMA, IOMMU. • Ataques físicos: troyanos de hardware, sondas de memoria, ataques de arranque en frío. • Seguridad de dispositivos integrados, por ejemplo, dispositivos médicos, automóviles. • Ruta confiable.	• Explica el concepto de integridad de código y firma de códigos, así como el alcance al cual se aplica [Familiarizarse] • Discute los conceptos del origen de la confidencialidad y el de los procesos de arranque y carga segura [Familiarizarse] • Describe los mecanismos de arresto remoto de la integridad de un sistema [Familiarizarse] • Resume las metas y las primitivas claves de los modelos de plataforma confiable (TPM) [Familiarizarse] • Identifica las amenazas de conectar periféricos en un dispositivo [Familiarizarse] • Identifica ataques físicos y sus medidas de control [Familiarizarse] • Identifica ataques en plataformas con hardware que no son del tipo PC [Familiarizarse] • Discute los conceptos y la importancia de ruta confiable [Familiarizarse]
Lecturas: W and L (2014)	

UNIDAD 9: Investigación digital (Digital Forensics) (25)	
Competencias: a,g	
Contenido	**Objetivos Generales**
• Principios básicos y metodologías de análisis digital forensico. • Diseñar sistemas con necesidades forenses en mente. • Reglas de Evidencia - conceptos generales y las diferencias entre las jurisdicciones y la Cadena de Custodia. • Búsqueda y captura de comprobación: requisitos legales y de procedimiento. • Métodos y normas de evidencia digital. • Las técnicas y los estándares para la conservación de los datos. • Cuestiones legales y reportes incluyendo el trabajo como perito. • Investigación digital de los sistema de archivos. • Los forenses de aplicación. • Investigación digital en la web. • Investigación digital en redes. • Investigación digital en dispositivos móviles. • Ataques al computador/red/sistema. • Detección e investigación de ataque. • Contra investigación digital.	• Describe qué es una investigación digital, las fuentes de evidencia digital, y los límites de técnicas forenses [Familiarizarse] • Explica como diseñar software de apoyo a técnicas forenses [Familiarizarse] • Describe los requisitos legales para usar datos recuperados [Familiarizarse] • Describe el proceso de recolección de evidencia desde el tiempo en que se identifico el requisito hasta la colocación de los datos [Familiarizarse] • Describe como se realiza la recolección de datos y el adecuado almacenamiento de los datos originales y de la copia forense [Familiarizarse] • Realiza recolección de datos en un disco duro [Usar] • Describe la responsabilidad y obligación de una persona mientras testifica como un examinador forense [Familiarizarse] • Recupera datos basados en un determinado término de búsqueda en una imagen del sistema [Usar] • Reconstruye el historial de una aplicación a partir de los artefactos de la aplicación [Familiarizarse] • Reconstruye el historial de navegación web de los artefactos web [Familiarizarse] • Captura e interpreta el tráfico de red [Familiarizarse] • Discute los retos asociados con técnicas forenses de dispositivos móviles [Familiarizarse]
Lecturas: W and L (2014)	

UNIDAD 10: Seguridad en Ingeniería de Software (25)	
Competencias: a,c,g,i	
Contenido	**Objetivos Generales**
• La construcción de la seguridad en el ciclo de vida de desarrollo de software. • Principios y patrones de diseño seguros. • Especificaciones de software seguros y requisitos. • Prácticas de desarrollo de software de seguros. • Asegure probar el proceso de las pruebas de que se cumplan los requisitos de seguridad (incluyendo análisis estático y dinámico)	• Describir los requisitos para la integración de la seguridad en el SDL [Familiarizarse] • Aplicar los conceptos de los principios de diseño para mecanismos de protección, los principios para seguridad de software (Viega and McGraw) y los principios de diseño de seguridad (Morrie Gasser) en un proyecto de desarrollo de software [Familiarizarse] • Desarrollar especificaciones para un esfuerzo de desarrollo de software que especifica completamente los requisitos funcionales y se identifican las rutas de ejecución esperadas [Familiarizarse]
Lecturas: W and L (2014)	

8. Metodología

El profesor del curso presentará clases teóricas de los temas señalados en el programa propiciando la intervención de los alumnos.

El profesor del curso presentará demostraciones para fundamentar clases teóricas.

El profesor y los alumnos realizarán prácticas

Los alumnos deberán asistir a clase habiendo leído lo que el profesor va a presentar. De esta manera se facilitará la comprensión y los estudiantes estarán en mejores condiciones de hacer consultas en clase.

9. Evaluar

Evaluación Continua 1 : 20 %

Examen parcial : 30 %

Evaluación Continua 2 : 20 %

Examen final : 30 %

References

W, Stallings. and Brown. L (2014). *Computer Security: Principles and Practice*. Pearson Education, Limited. ISBN: 9780133773927.