Introducción a la Programación Segura

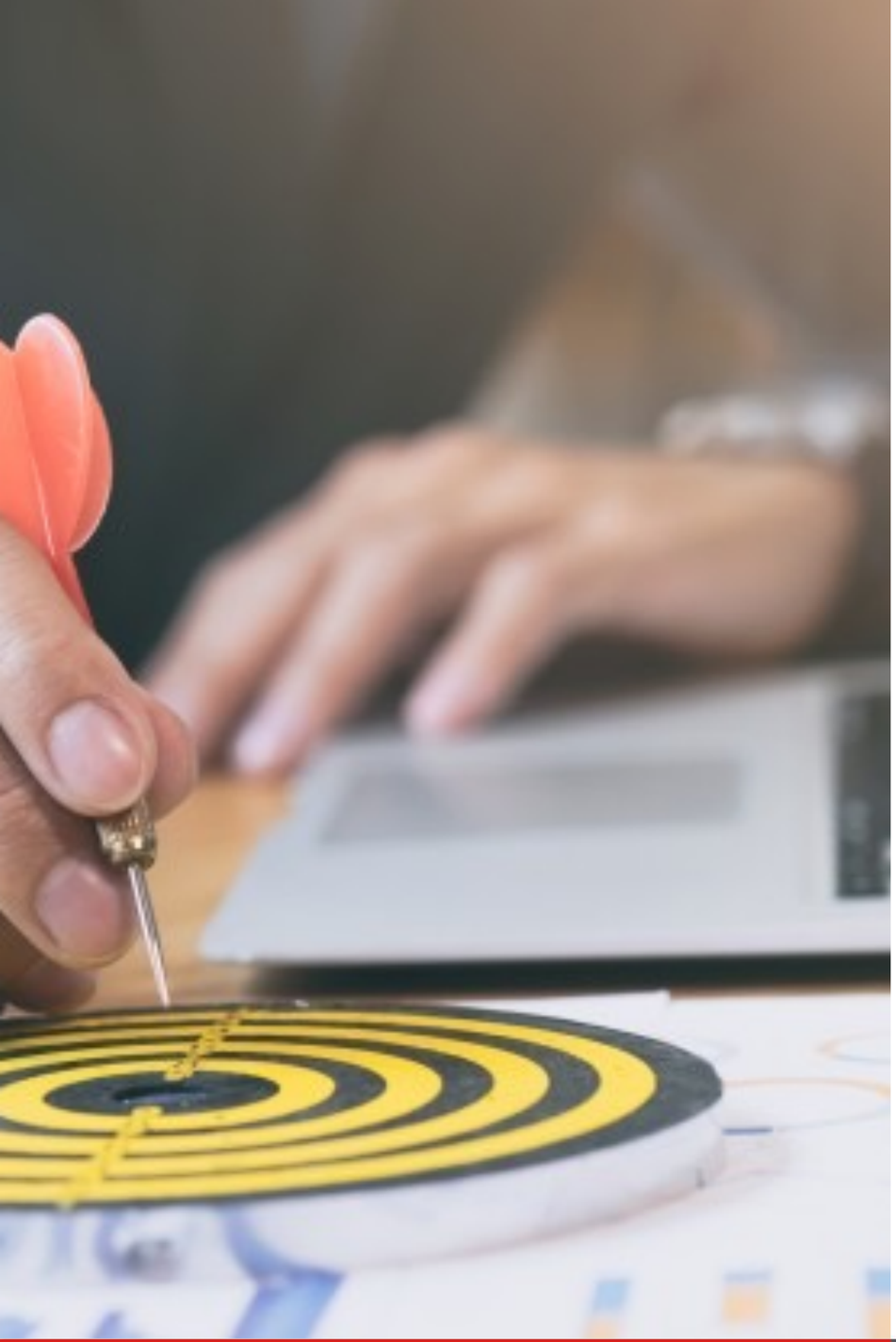
Colecciones y librerías en Python.

INTRODUCCIÓN A LA PROGRAMACIÓN SEGURA

UNIDAD 3: COLECCIONES Y LIBRERÍAS EN PYTHON.

Contenidos

1. Situaciones de Riesgo y Vulnerabilidades.
2. Abordando la Seguridad de los Datos.
3. Divulgación de Información Sensible.
4. Falta de Validación de Datos.



UNIDAD DE APRENDIZAJE

UNIDAD 3: COLECCIONES Y LIBRERÍAS EN PYTHON.

Aprendizaje Esperado:

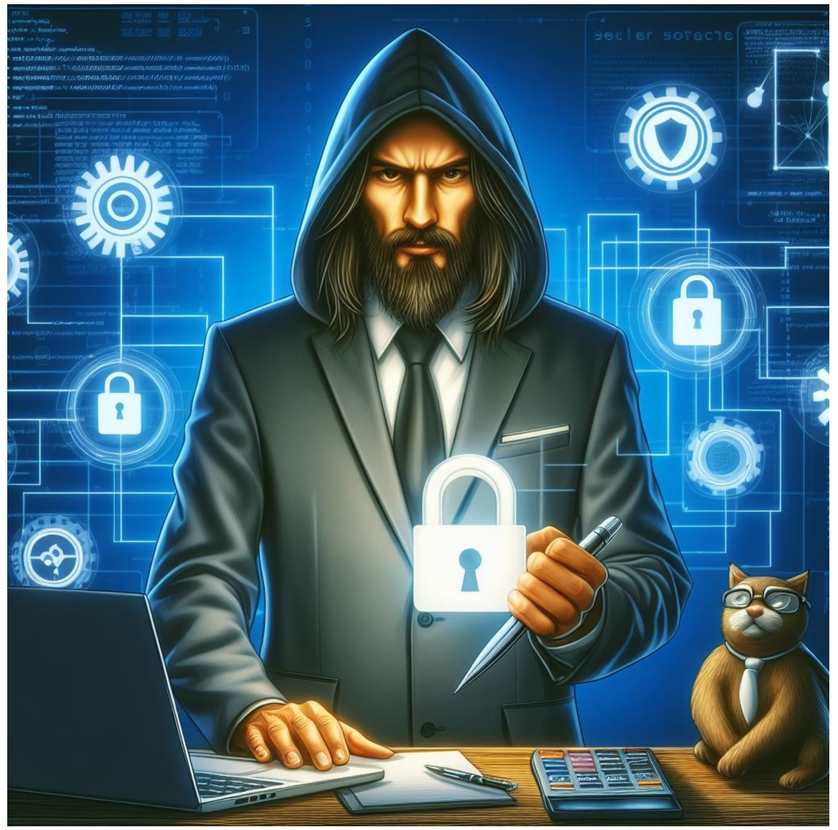
3.1. Utiliza estructuras de almacenamiento de datos de Python, para hacer más eficiente el código de programación, considerando el desarrollo de scripts y librerías asociadas a la seguridad.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

1. Situaciones de Riesgo y Vulnerabilidades

Algunas situaciones de riesgo y vulnerabilidades del almacenamiento de datos comunes en Python:

- Inyección de SQL .
- Cross-Site Scripting (XSS).
- Desbordamiento de búfer.
- Divulgación de información sensible.
- Falta de validación de datos.



Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

1. Situaciones de Riesgo y Vulnerabilidades

Inyección de SQL

La inyección de SQL es una vulnerabilidad común en la que un atacante puede manipular las consultas SQL de una aplicación enviando datos maliciosos a través de formularios web u otros canales de entrada. Esto puede llevar a la revelación de información confidencial o la eliminación no autorizada de datos en una base de datos.

Cross-Site Scripting (XSS)

El XSS es una vulnerabilidad que permite a los atacantes insertar scripts maliciosos en páginas web visitadas por otros usuarios. Estos scripts pueden robar cookies de sesión, redirigir a sitios web maliciosos o incluso modificar el contenido de la página web..

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

1. Situaciones de Riesgo y Vulnerabilidades

Desbordamiento de búfer

El desbordamiento de búfer ocurre cuando un programa intenta escribir más datos en un área de memoria reservada de lo que realmente puede contener. Esto puede llevar a la corrupción de datos, la ejecución de código malicioso o incluso al bloqueo del programa.

Divulgación de información sensible

La divulgación de información sensible ocurre cuando se exponen datos confidenciales, como contraseñas, claves de API o datos personales, a personas no autorizadas. Esto puede suceder debido a configuraciones incorrectas de permisos, errores de programación o fallos de seguridad en la aplicación.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

1. Situaciones de Riesgo y Vulnerabilidades

Falta de validación de datos

La falta de validación de datos puede permitir que los usuarios ingresen datos maliciosos o incorrectos en una aplicación. Esto puede conducir a errores de procesamiento, ataques de inyección de código y otros problemas de seguridad.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2. Abordando la Seguridad de los Datos

En esta etapa inicial de nuestra asignatura, nos centraremos en comprender y controlar dos riesgos fundamentales: la Divulgación de información sensible y la Falta de validación de datos. Dado nuestro nivel de introducción a la programación segura, es crucial establecer una base sólida en la seguridad de datos desde el principio

python™

inacap

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2. Abordando la Seguridad de los Datos

2.1. Divulgación de Información Sensible

La Divulgación de información sensible ocurre cuando datos confidenciales se exponen a personas no autorizadas. En nuestro curso, aprenderemos a identificar y proteger estos datos mediante prácticas seguras de manejo de información. Nos enfocaremos en la importancia de mantener la confidencialidad de contraseñas, claves de API y otros datos sensibles, así como en la implementación de medidas de seguridad apropiadas para evitar su divulgación.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.1. Divulgación de Información Sensible

Aplicando Medidas Control

Controlar el riesgo de Divulgación de Información Sensible implica implementar medidas de seguridad para proteger los datos confidenciales y prevenir su exposición a personas no autorizadas.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.1. Divulgación de Información Sensible– Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Clasificación de Datos: Identifica y clasifica los datos según su nivel de sensibilidad. Esto te ayudará a priorizar la protección de los datos más críticos y a asignar recursos de seguridad de manera efectiva.

Acceso Controlado: Implementa controles de acceso para limitar quién puede acceder a los datos sensibles. Utiliza autenticación y autorización para verificar la identidad de los usuarios y restringir el acceso solo a aquellos que tienen permiso.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.1. Divulgación de Información Sensible– Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Encriptación de Datos: Utiliza técnicas de encriptación para proteger los datos sensibles mientras están en tránsito y en reposo. Esto garantizará que incluso si los datos son interceptados, no puedan ser leídos sin la clave de encriptación adecuada.

Auditoría y Monitoreo: Establece sistemas de auditoría y monitoreo para realizar un seguimiento de quién accede a los datos sensibles y qué acciones realizan. Esto te permitirá detectar y responder rápidamente a cualquier actividad sospechosa o no autorizada.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.1. Divulgación de Información Sensible– Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Formación y Concientización: Capacita a los empleados sobre la importancia de proteger los datos sensibles y cómo hacerlo correctamente. Fomenta una cultura de seguridad en la organización para que todos comprendan su responsabilidad en la protección de la información confidencial.

Actualizaciones y Parches: Mantén tus sistemas y software actualizados con los últimos parches de seguridad para mitigar vulnerabilidades conocidas que podrían ser explotadas para acceder a datos sensibles.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.1. Divulgación de Información Sensible– Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Gestión de Incidentes: Desarrolla un plan de respuesta a incidentes para actuar rápidamente en caso de una brecha de seguridad o divulgación de datos sensibles. Esto incluye procedimientos para investigar, contener y mitigar los impactos de un incidente de seguridad.

Implementando estas estrategias y medidas de seguridad, puedes controlar de manera efectiva el riesgo de Divulgación de Información Sensible y proteger los datos confidenciales de tu organización. Es importante adoptar un enfoque proactivo hacia la seguridad de los datos y estar siempre atento a las nuevas amenazas y vulnerabilidades emergentes.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2. Abordando la Seguridad de los Datos

2.2. Falta de Validación de Datos

La Falta de validación de datos puede conducir a errores de procesamiento y vulnerabilidades de seguridad en nuestras aplicaciones. En nuestra asignatura, aprenderemos técnicas para validar y sanitizar los datos de entrada, asegurando que solo se acepten datos válidos y seguros. Exploraremos la importancia de la validación de datos en la prevención de ataques de inyección de código y otros vectores de ataque.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.2. Falta de Validación de Datos

Aplicando Medidas Control

Controlar el riesgo de Falta de Validación de Datos implica implementar mecanismos para garantizar que los datos ingresados en una aplicación sean correctos, seguros y cumplan con ciertos criterios predefinidos.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.2. Falta de Validación de Datos – Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Validación de Entrada: Implementa controles de validación en todos los puntos de entrada de datos de tu aplicación, como formularios web, entradas de usuario y solicitudes de API. Esto incluye verificar la longitud, el formato, el tipo de datos y la ausencia de caracteres especiales inesperados.

Sanitización de Datos: Lleva a cabo la sanitización de datos para eliminar cualquier carácter peligroso o inseguro que pueda ser utilizado en ataques de inyección de código, como SQL injection o Cross-Site Scripting (XSS). Esto implica escapar o eliminar caracteres especiales y codificar correctamente los datos antes de almacenarlos o procesarlos.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.2. Falta de Validación de Datos – Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Validación del Lado del Servidor: Realiza la validación de datos en el lado del servidor para garantizar que los datos enviados por el cliente sean válidos y seguros. No confíes únicamente en la validación del lado del cliente, ya que puede ser fácilmente eludida por un atacante.

Validación de Longitud y Formato: Verifica que los datos ingresados cumplan con los requisitos de longitud y formato especificados. Esto puede incluir la validación de direcciones de correo electrónico, números de teléfono, fechas, contraseñas, entre otros.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.2. Falta de Validación de Datos – Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Validación de Rangos y Valores Permitidos: Asegúrate de que los datos ingresados estén dentro de rangos válidos y cumplan con los valores permitidos. Esto ayuda a prevenir errores de procesamiento y evita que los usuarios ingresen datos incorrectos o maliciosos.

Gestión de Errores y Mensajes de Error Seguros: Proporciona mensajes de error claros y seguros para informar a los usuarios sobre problemas de validación. Evita revelar información sensible en los mensajes de error y asegúrate de que los mensajes sean útiles para ayudar a los usuarios a corregir los errores.

Situaciones de riesgo y vulnerabilidades del almacenamiento de datos, según estándar de seguridad de Python:

2.2. Falta de Validación de Datos – Aplicando Medidas Control

Veamos algunas estrategias clave para controlar este riesgo:

Pruebas de Penetración y Auditorías de Seguridad: Realiza pruebas de penetración y auditorías de seguridad periódicas para identificar posibles vulnerabilidades de falta de validación de datos en tu aplicación. Esto te ayudará a detectar y corregir problemas de seguridad antes de que sean explotados por atacantes.

Si implementamos estas estrategias y prácticas de validación de datos, puedes controlar de manera efectiva el riesgo de Falta de Validación de Datos y garantizar la integridad y seguridad de los datos en tu aplicación. Es importante adoptar un enfoque proactivo hacia la validación de datos y estar siempre atento a las nuevas amenazas y vulnerabilidades en el panorama de seguridad cibernética.

inacap.cl