Programación con IA Generativa

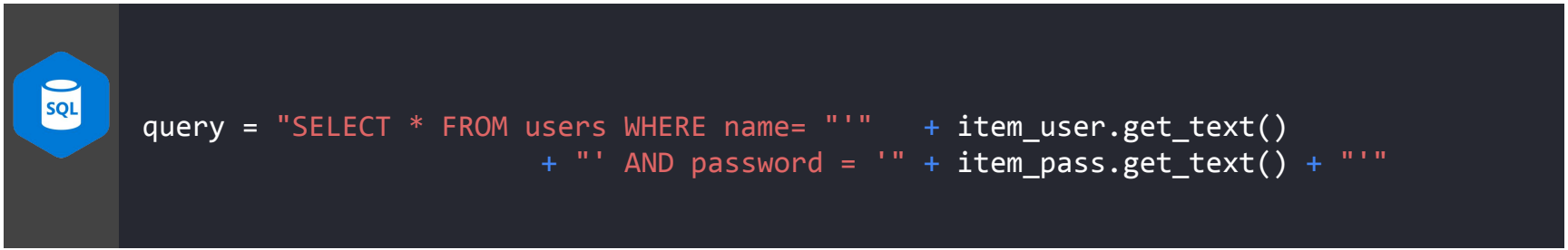
Seguridad con IA

Búsqueda de vulnerabilidades

Aunque muchos desarrolladores pueden considerarlos "conocimiento común", la gran mayoría de los puntos débiles de seguridad se deben a vulnerabilidades como el **scripting entre sitios** (XSS), la **inyección de código SQL** y la **falsificación de solicitudes entre sitios** (CSRF).

Estas vulnerabilidades se pueden mitigar mediante prácticas de codificación segura, como usar consultas con parámetros, validar las entradas y evitar datos confidenciales codificados de forma rígida, donde nos puede ayudar un asistente de IA.

Ejemplo: Inyección de código SQL

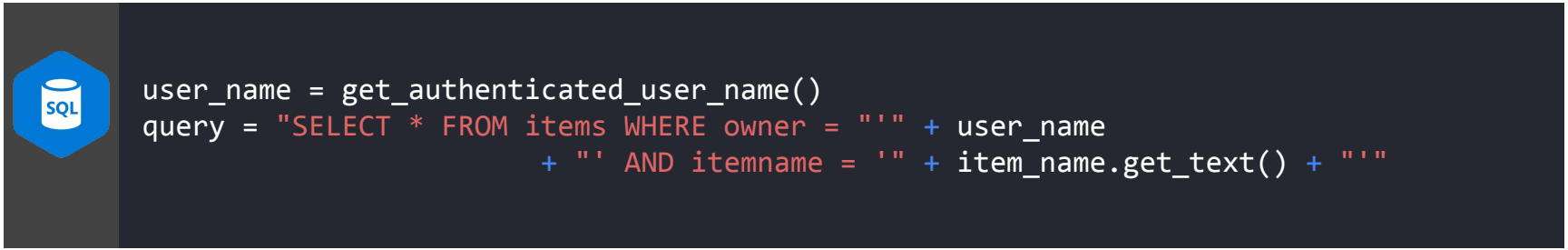


Este código presenta una vulnerabilidad: el uso de la concatenación.
Si el atacante proporciona el siguiente texto para item_pass: **sssss' OR 5=5** Tras combinar el nombre de usuario y el nombre del elemento, el código crea la siguiente consulta :

SELECT * FROM users
WHERE user= '' AND password= 'sssss' OR 5=5';

Esto significa que la consulta devolverá los datos de toda la tabla, lo que le dará al atacante acceso no autorizado a datos confidenciales.

Ejemplo: Inyección de código SQL



Este código presenta una vulnerabilidad: el uso de la concatenación.
Si el atacante proporciona el siguiente texto para item_name: **sssss' OR 5=5** Tras combinar el nombre de usuario y el nombre del elemento, el código crea la siguiente consulta :

SELECT * FROM items
WHERE owner = 'John'
AND itemname = 'sssss' OR 5=5';

Esto significa que la consulta devolverá los datos de toda la tabla, lo que le dará al atacante acceso no autorizado a datos confidenciales.

Consideraciones a tener en cuenta

Podemos solicitar al asistente de IA que analice el código para detectar vulnerabilidades de seguridad comunes y que proporcione explicaciones y correcciones para los problemas que encuentre. Por ejemplo:

Analizar el código por posibles potenciales de vulnerabilidad y sugerir correcciones.

Ejemplo: Inyección de código SQL


```
import mysql.connector

con=mysql.connector.connect(user="root",password="",host="127.0.0.1",database="ej")
cursor=con.cursor()

owner = user_name
itemname = item_name.get_text()
cursor.execute("SELECT * FROM items WHERE owner = ? AND itemname =?", owner, itemname))

con.commit()
con.close()
```

Actividad en grupo

Armar un ejemplo para documentar un código ya heredado, la explicación de un código en un lenguaje desconocido usando analogías con un lenguaje conocido o la explicación de un algoritmo complejo.

Armar otro ejemplo para prevenir una vulnerabilidad de seguridad.

Actividad en grupo: 20:12 a 20:30
Sala principal: 20:30