**Lab: SQL injection vulnerability in WHERE clause allowing retrieval of hidden data**

This lab contains a [SQL injection](https://portswigger.net/web-security/sql-injection) vulnerability in the product category filter. When the user selects a category, the application carries out a SQL query like the following:

***SELECT \* FROM products WHERE category = 'Gifts' AND released = 1***

To solve the lab, perform a SQL injection attack that causes the application to display one or more unreleased products.

La solución es colocar como parametro de category una sentencia que modifique la consulta SQL ejecutada por la URL.

***'+OR+1=1—***

+: en la expresión '+OR+1=1-- no forma parte de la sintaxis de SQL. En realidad, es una codificación utilizada en las URLs. Específicamente, el + es un espacio codificado en URLs cuando se envían solicitudes HTTP.

****** ': Cierra una cadena de texto que puede haber sido iniciada en la consulta original.

 OR: Introduce una condición lógica. En este caso, se está añadiendo un OR a la cláusula WHERE para forzar a que la consulta devuelva resultados.

 1=1: Es una expresión que siempre es verdadera. Al añadir esta expresión con el operador OR, se asegura que la consulta siempre devuelva todas las filas, porque cualquier condición en la consulta se evaluará como verdadera.

 --: Es un comentario en SQL. Todo lo que viene después de -- se ignora. Esto permite al atacante eliminar el resto de la consulta original que podría restringir los resultados.

