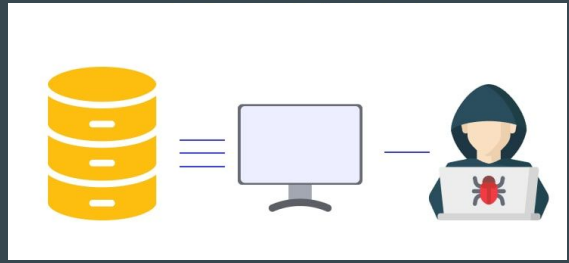
NoSQL Injection

Seguridad e Integridad de Sistemas



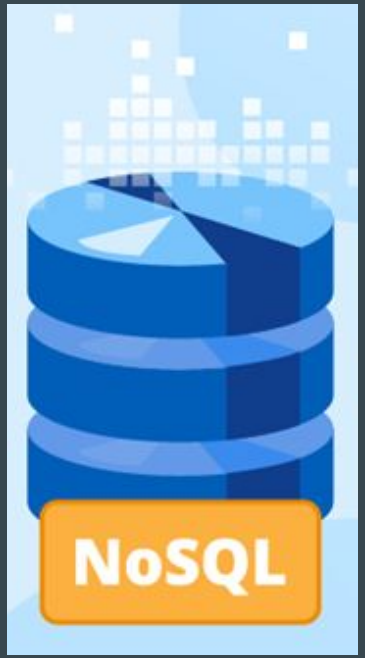
Prof. GIANNI, Fernando Julian

¿Qué es una inyección NoSQL?

La inyección NoSQL es una técnica de ataque donde un atacante manipula consultas NoSQL a través de entradas no validadas.

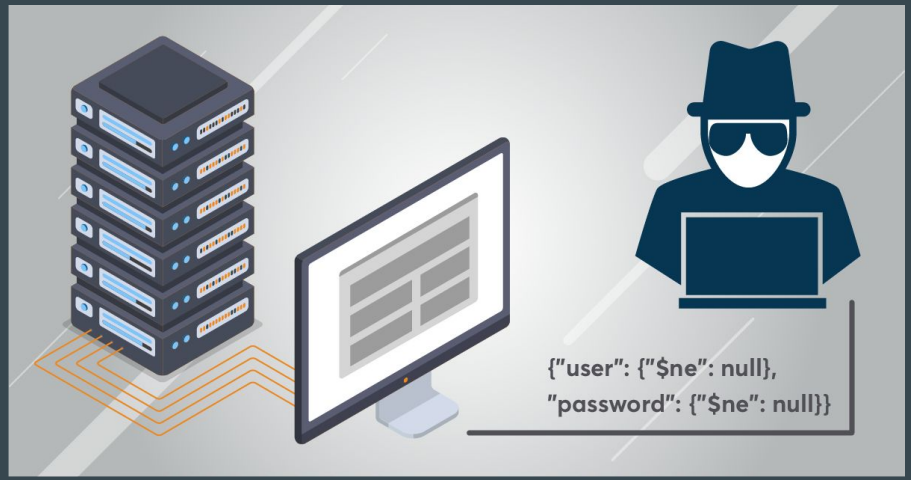
Similar a la inyección SQL, pero aplicada a bases de datos NoSQL, que son populares en aplicaciones modernas por su flexibilidad y escalabilidad.

Permite a los atacantes acceder, modificar o borrar información no autorizada.



¿Cómo se explota?

- Entrada no validada: El atacante envía entradas maliciosas a través de formularios, URLs o parámetros de API.
- Manipulación de la consulta: El sistema construye la consulta NoSQL utilizando los datos de entrada sin validación o sanitización, permitiendo la inyección.
- Ejemplo:
  - Parámetro de URL: ?category=Gifts||1||
  - Consulta original: { category: "Gifts" }
  - Consulta manipulada: { $or: [{ category: "Gifts" }, { 1: 1 }] }, devolviendo todos los productos.

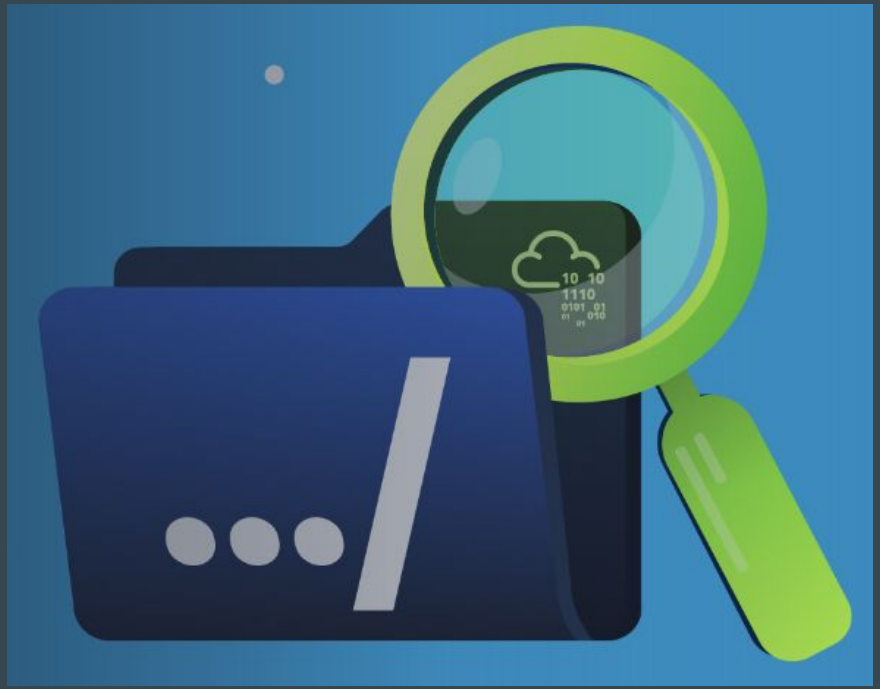


¿Cómo encontrarla?

- Testing manual:
  - Intentar manipular los parámetros de entrada con operadores como ||, $gt, $lt, o caracteres especiales.
  - Revisar las respuestas del servidor para identificar comportamientos anómalos.
- Herramientas de análisis:
  - Utilizar herramientas de pentesting automatizado como Burp Suite, que permiten detectar inyecciones NoSQL.
- Revisar la lógica de consultas:
  - Buscar casos donde las entradas del usuario son usadas directamente en consultas sin validación.

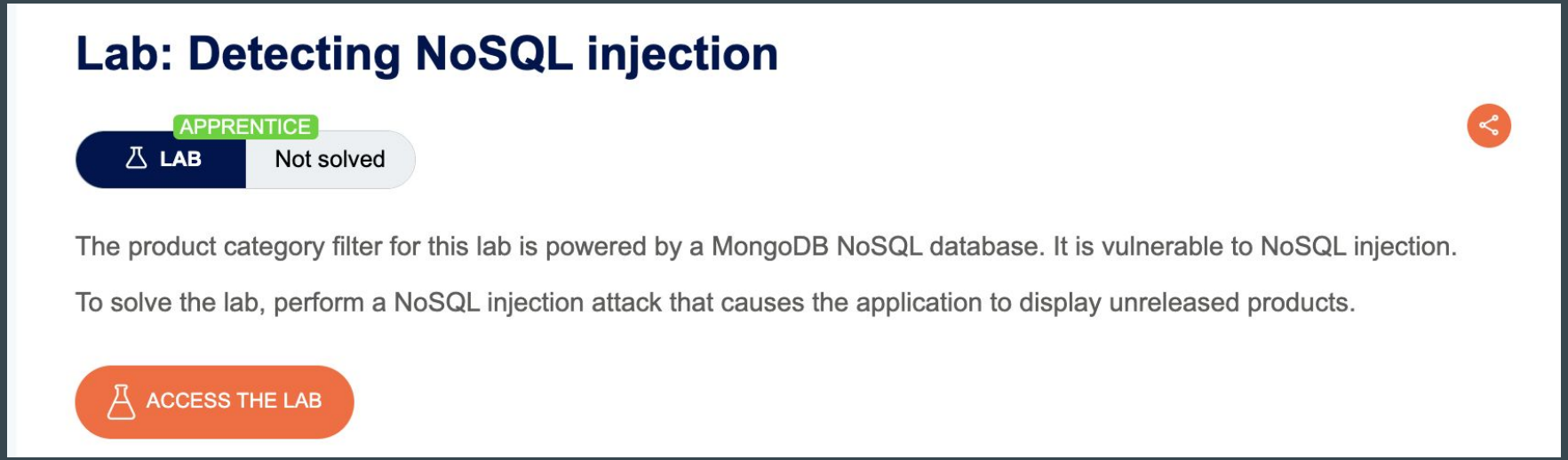
Riesgo

1) Acceso no autorizado a datos: Un atacante puede leer o modificar datos sensibles de la base de datos.
2) Exposición completa de la base de datos: Los datos críticos de la empresa, como información de usuarios, transacciones o inventarios, pueden ser expuestos.
3) Escalación de privilegios: Un atacante puede elevar sus privilegios dentro de la aplicación o acceder a funciones restringidas.
4) Modificación o eliminación de datos: Impacto en la integridad de los datos almacenados.



Laboratorios

https://portswigger.net/web-security/nosql-injection/lab-nosql-injection-bypass-authentication



Mitigación

- Validación y sanitización de entradas:
  - Asegurarse de que todos los datos de entrada sean estrictamente validados antes de usarse en una consulta.
- Uso de consultas parametrizadas:
  - Evitar la construcción dinámica de consultas; utilizar consultas parametrizadas siempre que sea posible.
- Limitación de operadores en las consultas:
  - Restringir el uso de operadores $or, $ne, $where, y otros que podrían ser manipulados.
- Principio de privilegios mínimos:
  - Limitar los permisos de las cuentas de la base de datos para que solo accedan a lo necesario.
- Monitoreo y auditoría:
  - Implementar un sistema de logs para detectar actividades sospechosas en la base de datos.