Universidad Corporativa | Programas Académicos

Ingeniería en Desarrollo de Software

Actividad 2 – Deserialización insegura

Auditoria informática

Ingeniería en Desarrollo de Software

Tutor:

Jessica Hernández Romero

Alumno:

Alejandro Abarca Gerónimo

Fecha:

22 de febrero de 2025

Indice

Introducción

La deserialización es el proceso de restaurar flujos de bytes a una réplica completamente funcional del objeto o credencial de acceso original, en el mismo estado en el que se encontraba cuando se serializó. Un sitio web puede utilizar la lógica para interactuar con el objeto deserializado. Dicha deserialización insegura se produce cuando un sitio web deserializa datos que el usuario puede controlar. Esto permite potencialmente que un atacante manipule objetos serializados para pasar datos dañinos al código de la aplicación.

Incluso es posible reemplazar un objeto serializado por un objeto de una clase completamente diferente. La deserialización insegura también se conoce como vulnerabilidad de "inyección de objetos".

Muchos ataques basados en la deserialización se completan antes de que la deserialización haya terminado. Esto significa que el proceso de deserialización en sí mismo puede iniciar un ataque, incluso si la propia funcionalidad del sitio web no interactúa directamente con el objeto malicioso. Por este motivo, los sitios web cuya lógica se basa en lenguajes fuertemente tipados también pueden ser vulnerables a estas técnicas.

Cuando se trata de violación de datos, las consecuencias son grandes y profundamente impactantes. Estas violaciones se han convertido en problemas de seguridad cibernética, así como pérdidas financieras, daños de reputación, problemas legales, multa regulatoria y la prevención de una profunda confianza del consumidor. A pesar del mayor peso dado a la seguridad de los datos, las personas dedicadas a la información encuentran nuevas formas de dividir para acceder a datos valiosos y datos de acreditación para las empresas. La pérdida de datos puede afectar a los usuarios de Internet de varias maneras, como el robo de identidad, la compensación de identidad y la pérdida financiera.

Descripción

Para el desarrollo de esta segunda actividad haremos el ataque a un sitio web para obtener las credenciales por medio de la deserialización insegura mediante el acceso a las cookies. Ya que una empresa de software solicita realizar varias pruebas de seguridad en sus páginas web que no cuentan con los candados de seguridad.

Para lograrlo, utilizar el programa Burp Suite Community Edition. El objetivo de esta prueba es que se inicie sesión como un usuario normal y luego pasar a modo administrador a través de las cookies.

También utilizaremos la plataforma PortSwigger, para realizar dicho ataque a una la página proporcionada. En ella, iniciaremos sesión con las credenciales que se nos proporcionaron, las cuales son para usuarios normales; no obstante, a través de las cookies, entraremos al modo administrador.

El objetivo es eliminar la cuenta de Carlos.
Hay que iniciar sesión en la propia cuenta con las siguientes credenciales:

- **Usuario:** wiener
- **Contraseña**: peter

Justificación

El ataque de pérdida de autenticación de datos genera graves consecuencias para los usuarios de Internet. Este tipo de ataque se refiere a la situación en la que un atacante gana acceso no autorizado a las credenciales de un usuario (como contraseñas, tokens de sesión, o datos biométricos), lo que puede permitirle acceder a servicios, cuentas y recursos personales.

Las auditorías informáticas son eficaces porque ayudan a evaluar el estado de los sistemas de información de una organización, identificando áreas de mejora

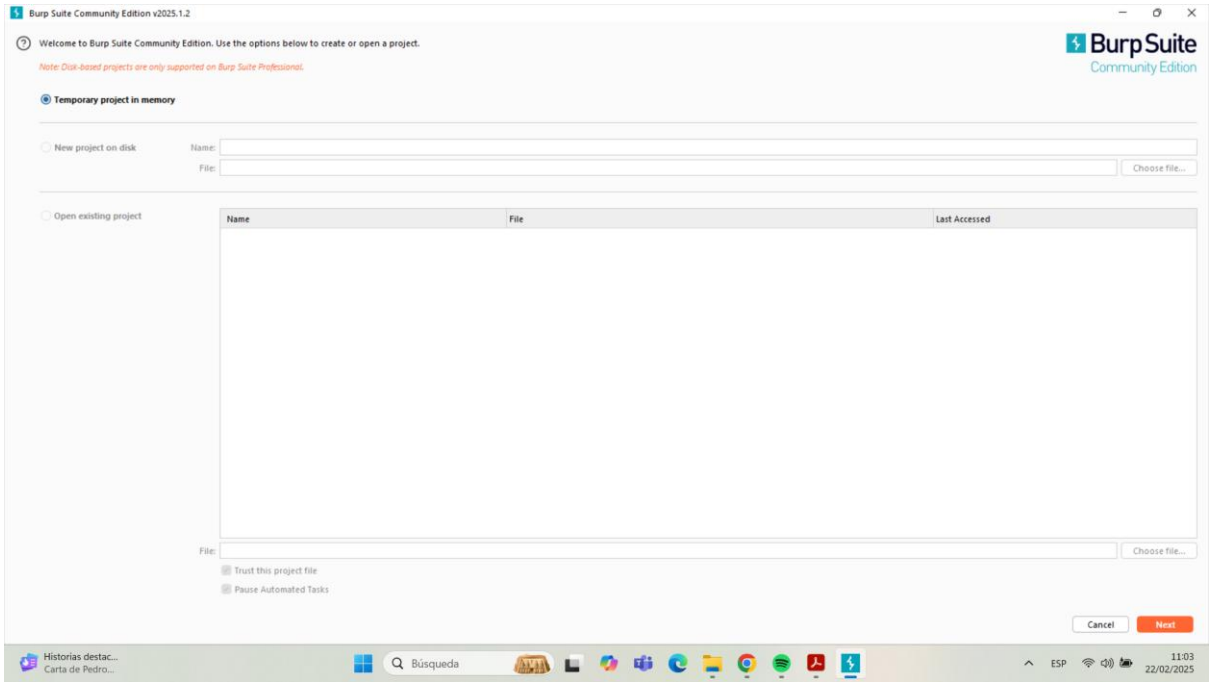
y asegurando que los recursos tecnológicos se utilicen de manera eficiente y segura.

Estas son algunas de las principales tareas a realizar en una auditoría informática:

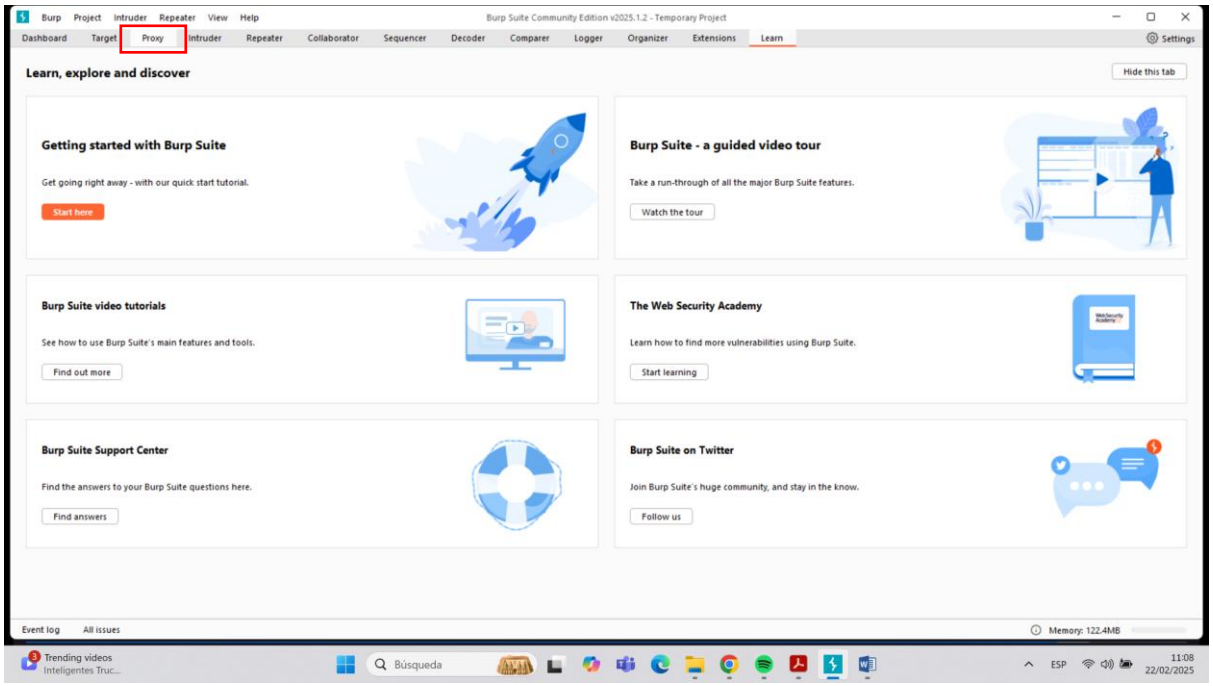
- Seguridad de la información.
- Cumplimiento normativo.
- Riesgos tecnológicos.
- Eficiencia operativa.
- Gestión de cambios.
- Controles internos y procedimientos.
- Planes de contingencia y recuperación.
- Responsabilidades y roles.

Ataque al sitio

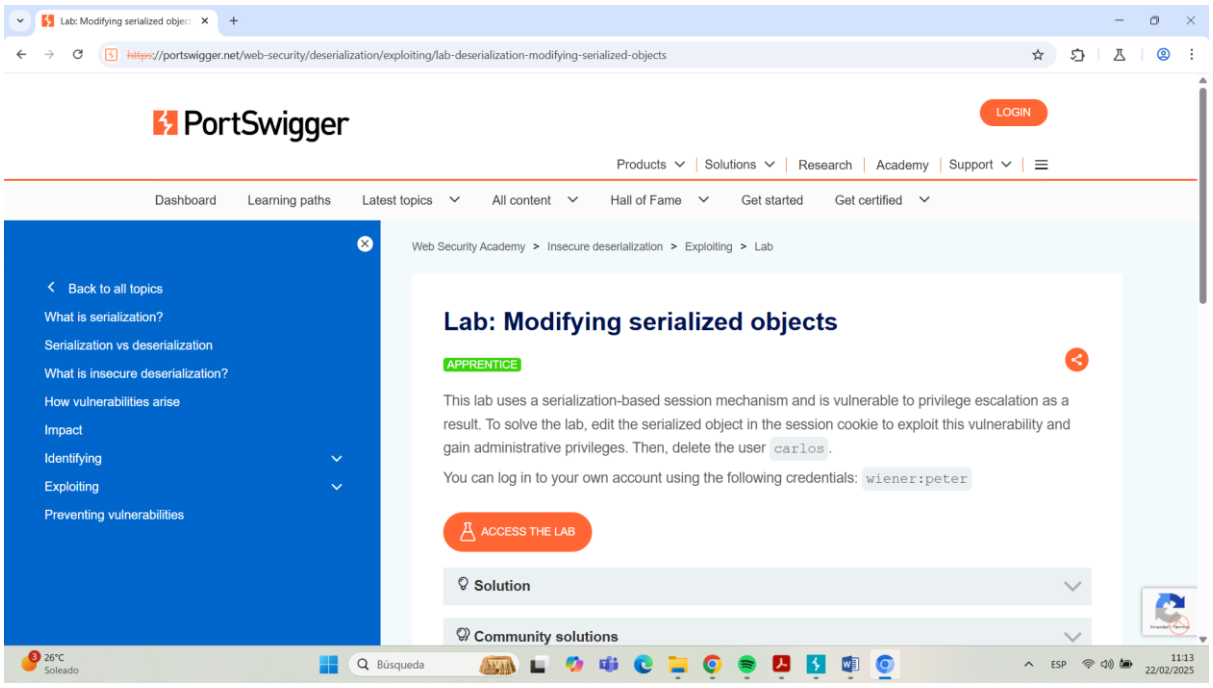
Como primer paso ingresamos al link de descarga proporcionado en el documento de la actividad para descargar la herramienta solicitada y procedimos a realizar la instalación.



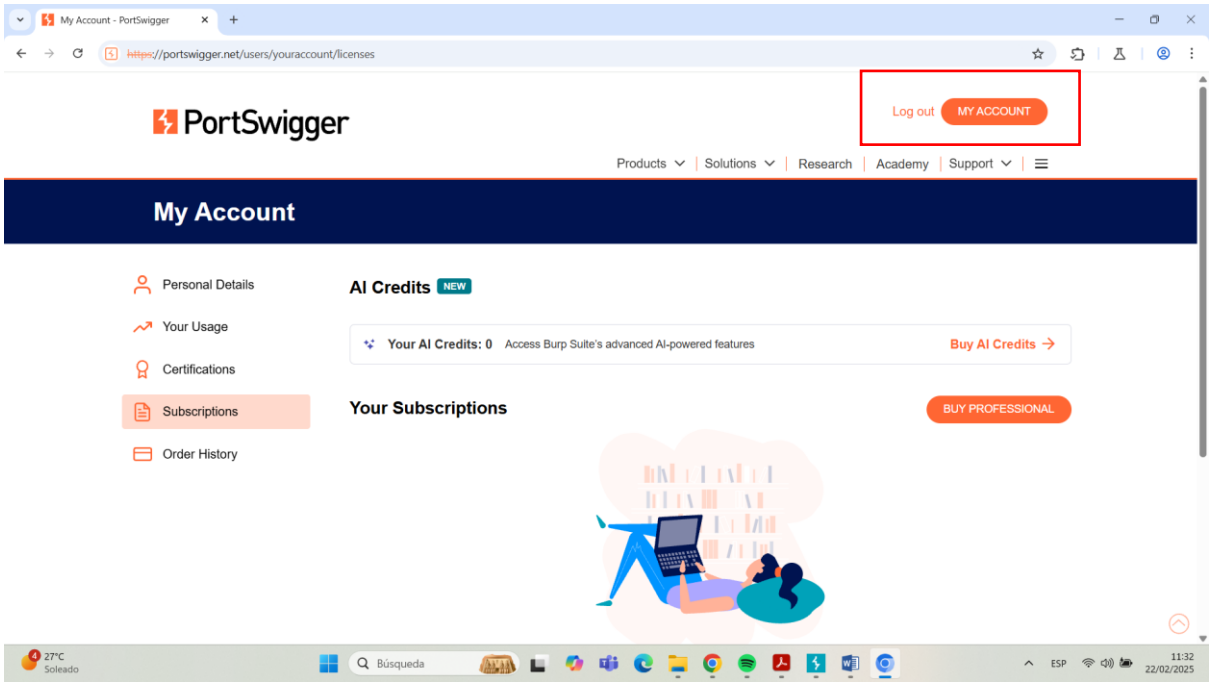
Una vez creado el proyecto y realizar las configuraciones correspondientes se procede a realizar el proceso que se solicita en la actividad ingresando a la pestaña PROXY.



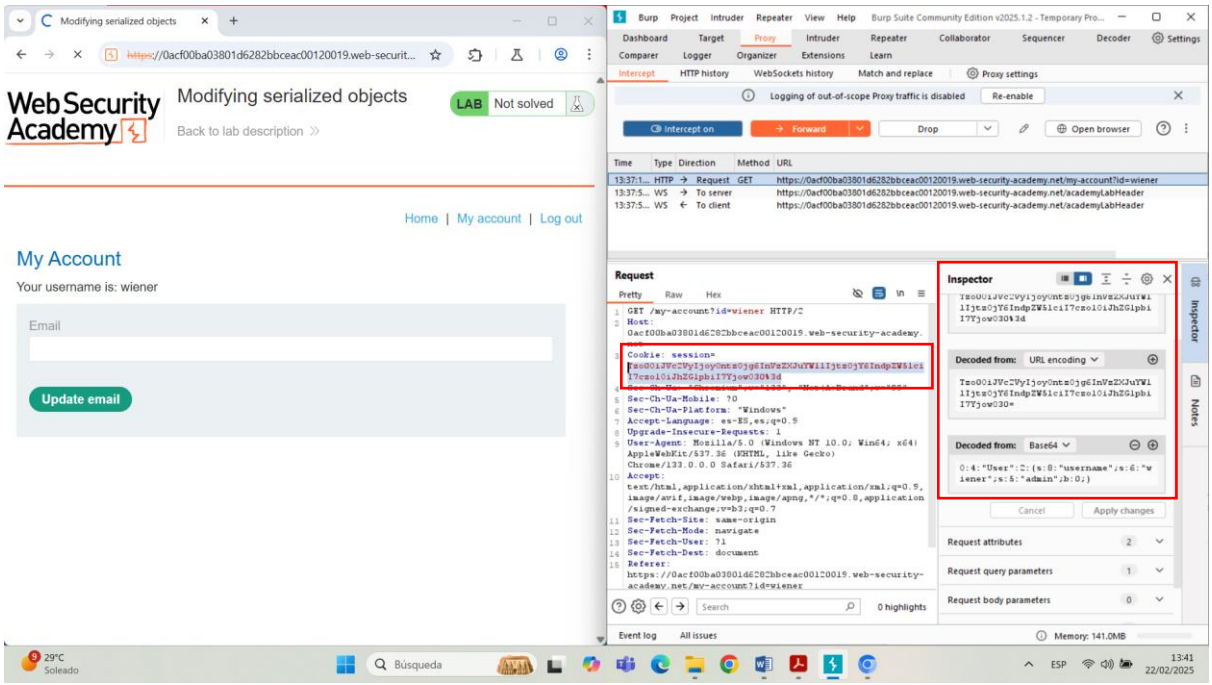
Después de haber ingresado a la pestaña antes mencionada, procedemos a abrir el Browser el cual es el navegador de la herramienta a utilizar y colocamos el link del laboratorio para trabajar dentro de este navegador.



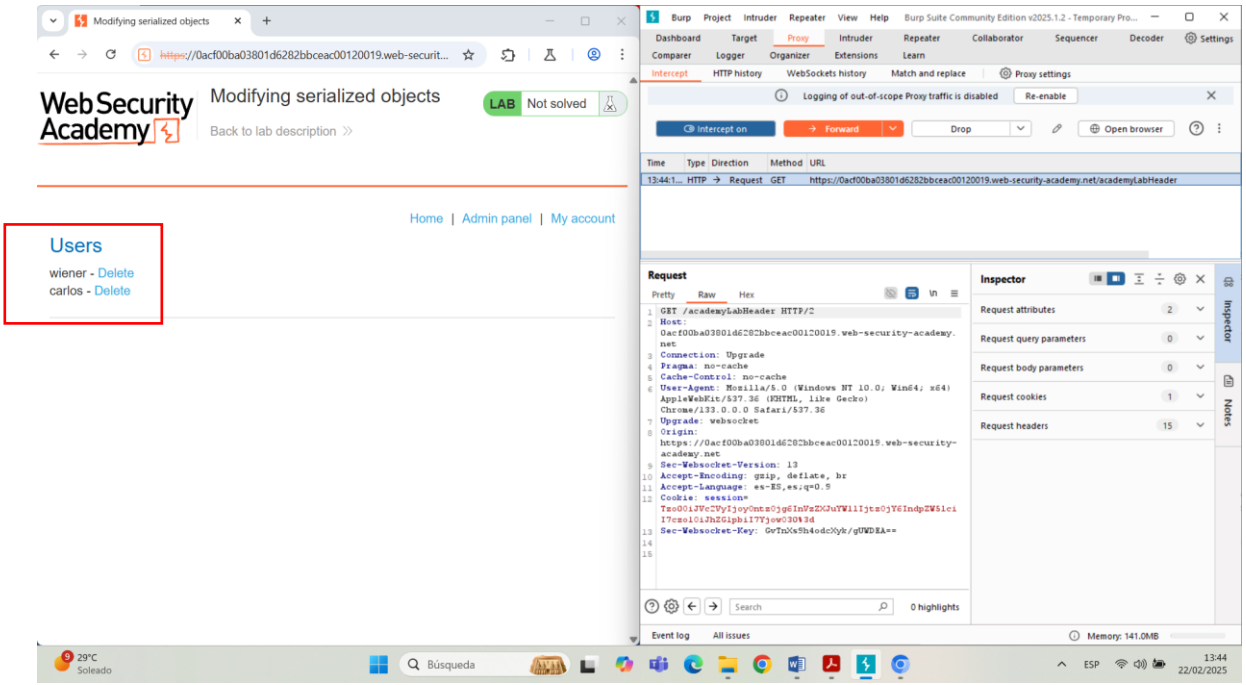
Una vez que ingresamos a la cuenta creada con anterioridad durante la descarga de Burp Suite, ingresamos al apartado MY ACCOUNT para ingresar las credenciales proporcionadas en el archivo de la actividad.



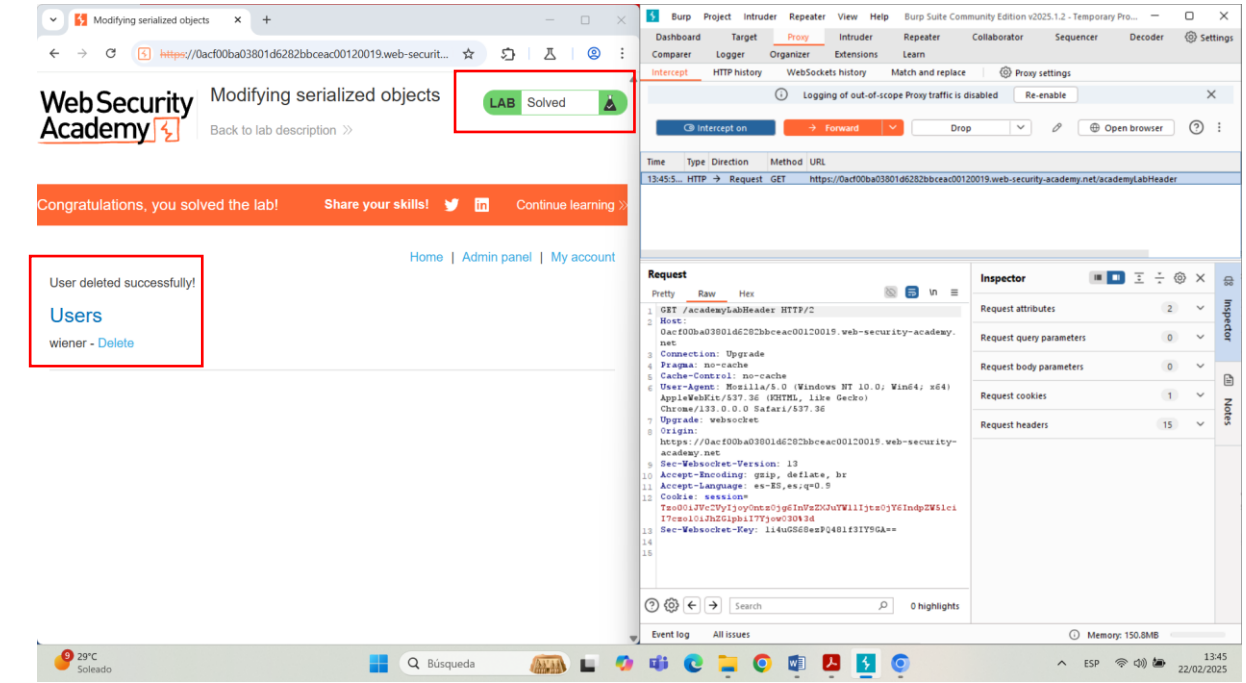
Una vez que ingresamos a la cuenta proporcionada encendemos el **Intercept on** para poder visualizar la cookie, seleccionamos y visualizamos en el apartado inspector para poder habilitar el modo admin. con valor de 1.



Una vez realizado dicho cambio de valor y pasar a tener derechos de administrador podemos visualizar los usuarios, en este caso el que tenemos que borrar es el usuario **Carlos.** Damos clic en **Delete** y nuevamente seleccionamos la cookie para modificar el valor de 0 a 1 y demostrar que somos **admin.**



En la siguiente pantalla se muestra el ejercicio resuelto con la cuenta de **Carlos** eliminada, sólo queda el usuario **wiener**, usuario del cual se nos proporcionaron las credenciales.



Conclusión

Durante el desarrollo de esta segunda actividad aprendimos el significado de la deserialización insegura, este es un riesgo de seguridad que ocurre cuando una aplicación convierte datos recibidos de una fuente externa en un objeto o estructura de datos para usar como accesos y permite tener el control sobre los mismos administradores. Dicho proceso de deserialización permite transformar una cadena de texto o un flujo de bytes en un objeto en memoria que la aplicación puede usar. Para evitar el robo de información es importante que se tomen precauciones adecuadas, para que los atacantes no puedan manipular los datos enviados, y no logren que el proceso de deserialización ejecute código malicioso o cause comportamientos inesperados en los sistemas.

La deserialización insegura se convierte en un riesgo porque los datos deserializados pueden ser manipulados por los atacantes de tal manera que pueda modificar los datos para incluir instrucciones o funciones maliciosas que se ejecuten durante el proceso de deserialización. También los atacantes pueden tener acceso a recursos no autorizados si los objetos deserializados contienen referencias a recursos sensibles como archivos del sistema, bases de datos, etc.

La deserialización insegura es una vulnerabilidad conocida y explotada en diversas aplicaciones. Por lo tanto, se debe tener cuidado al manejar datos provenientes de fuentes externas para prevenir este tipo de ataques.

Referencias

1. *Insecure deserialization | Web Security Academy*. (s. f.). https://portswigger.net/web-security/deserialization
2. Strawbridge, G. (2024, 2 septiembre). Data breach: 5 graves consecuencias de la violación de datos. *MetaCompliance*. https://www.metacompliance.com/es/blog/data-breaches/5-damaging-consequences-of-a-data-breach
3. Ponce, J. L. (2023, 9 agosto). *Auditoría informática: ¿Qué es y cómo hacer una con éxito?* Ikusi MX. https://www.ikusi.com/mx/blog/auditoria-informatica/