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: Gabriel German Verdugo Solís

Fecha: 17 de febrero del 2024

INDICE

INTRODUCCION

La autentificación de datos es el procedimiento informático a través del que aseguramos que el usuario de un sitio o servicio es quien dice ser. La autenticación de datos constituye uno de los principales pilares de la gestión del control de accesos, junto a la trazabilidad y la autorización. Cualquier sistema de control de accesos que quiera garantizar la seguridad de la empresa y de sus recursos necesita tomar conciencia de la importancia de la autenticación. El control de acceso y autenticación son acciones destinadas a proteger los sistemas informáticos, como los servidores y los ordenadores. Su objetivo es dotar a los negocios de mecanismos con los que gestionar usuarios y datos de identificación, así como controlar el acceso a los recursos. Los ataques en los sistemas de autenticación se consideran como el segundo procedimiento más usado para vulnerar sistemas. La prevalencia de este tipo de ataque está muy extendida debido al diseño e implementación de la mayoría de los controles de identidad y acceso. La administración de sesiones es la base de los controles de autenticación y está presente en todas las aplicaciones. Los atacantes pueden detectar este tipo de vulnerabilidad utilizando medios manuales y explotarlos utilizando herramientas automatizadas con listas de contraseñas y ataques de diccionario. La mayoría de los ataques de autenticación se producen debido al uso continuo de contraseñas como único factor. Las políticas de rotación y los requisitos de complejidad en las contraseñas que no han sido bien diseñados son causantes de este tipo de vulnerabilidad, entre otras.

DESCRIPCION

En esta actividad se aprenderá a realizar un ataque a una página con la ayuda de portswigger en ella se iniciará sesión con las credenciales que se proporcionan y a través de las cookies se entrara en modo administrador en este laboratorio se utiliza un mecanismo de sesión basada en serialización, por ende, es vulnerable a la escalada de privilegios. En consecuencia, hay editar el objeto serializado en la cookie de sesión para aprovechar esta vulnerabilidad y obtener privilegios administrativos. Finalmente, el objetivo es eliminar la cuenta de Carlos.

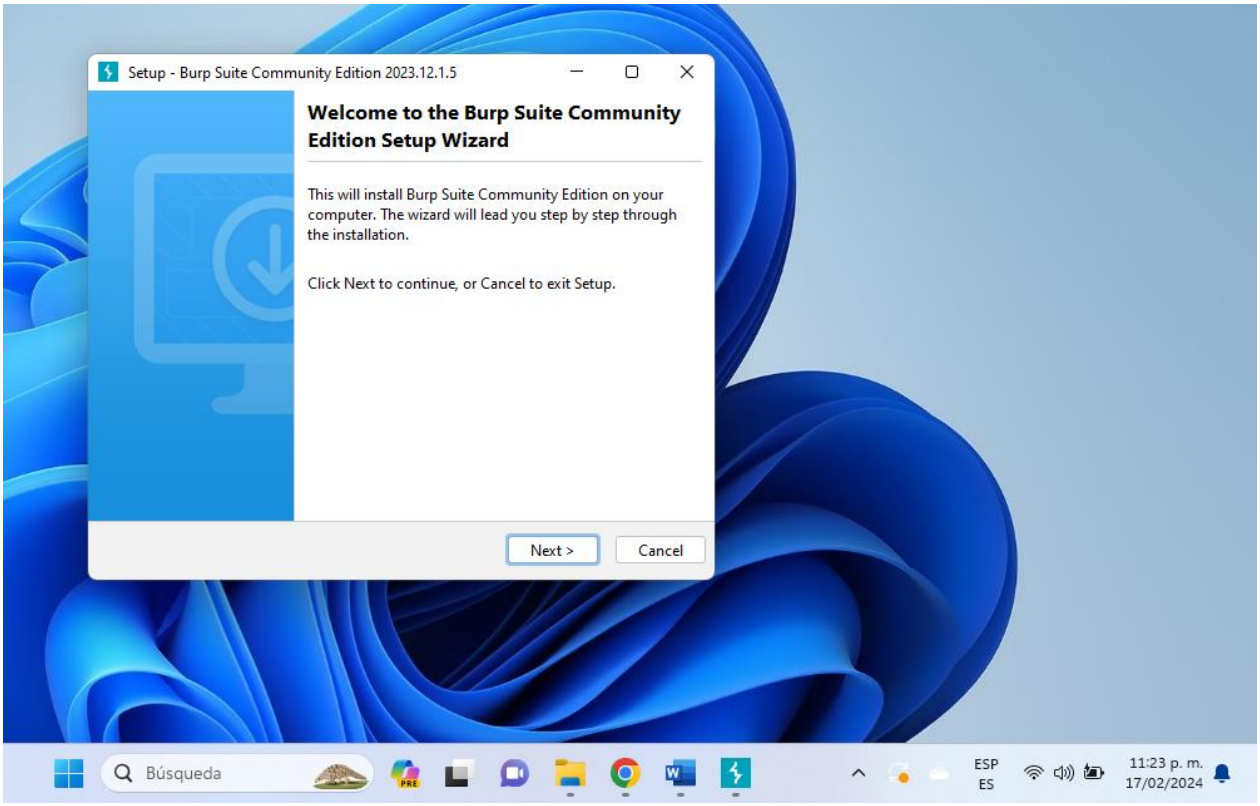
JUSTIFICACION

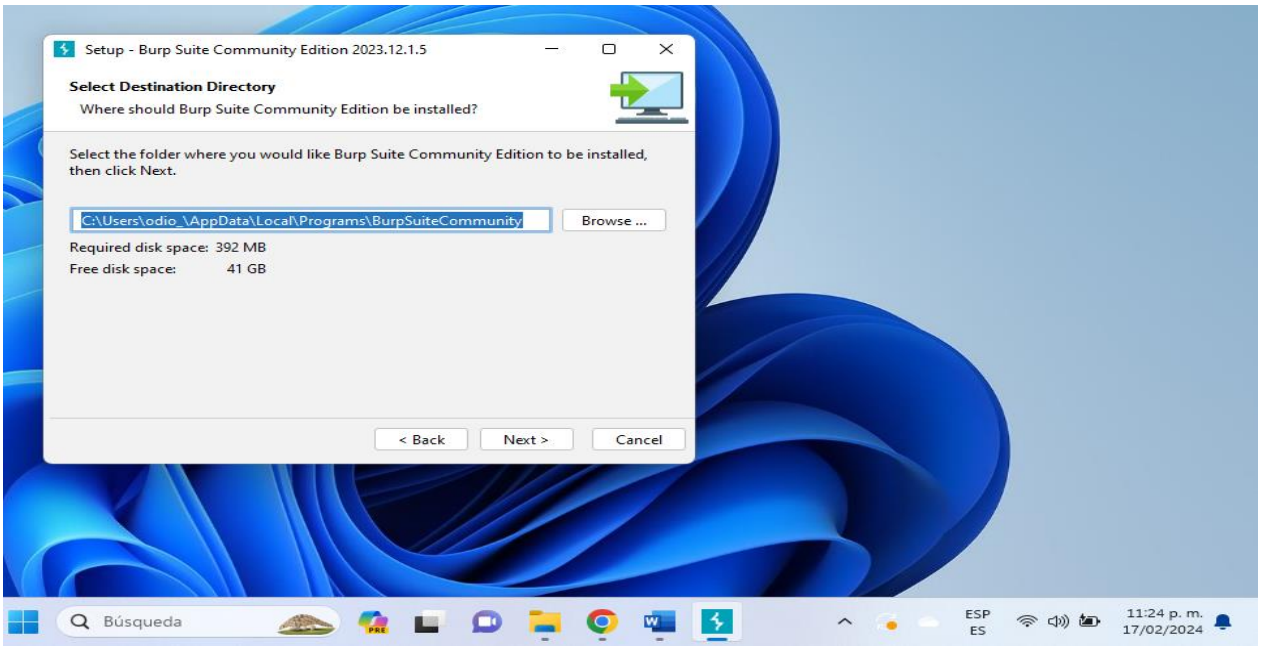
La deserialización insegura es una vulnerabilidad que ocurre cuando una aplicación o una API deserializa datos manipulados por un atacante en el lado del servidor. Es decir, durante el momento en el que se transforman los datos serializados a un objeto, un atacante puede abusar de la lógica de la aplicación, y realizar ataques de denegación de servicio, omitir autenticaciones o incluso ejecutar código malicioso de forma remota. Generalmente esta vulnerabilidad ocurre por falta de comprensión de lo peligroso que puede ser la deserialización de datos controlables por el usuario. Ya que, idealmente, la entrada del usuario nunca debería deserializarse en absoluto. Es decir que no es posible deserializar de forma segura una entrada que no es de confianza. La deserialización insegura permite a un atacante manipular objetos serializados para pasar datos dañinos al código de la aplicación, e incluso, reemplazar un objeto serializado por un objeto de una clase distinta. Es decir, los objetos que estén disponibles para el sitio web serán deserializados e instanciados, independientemente de la clase que se esperaba. De hecho, es por

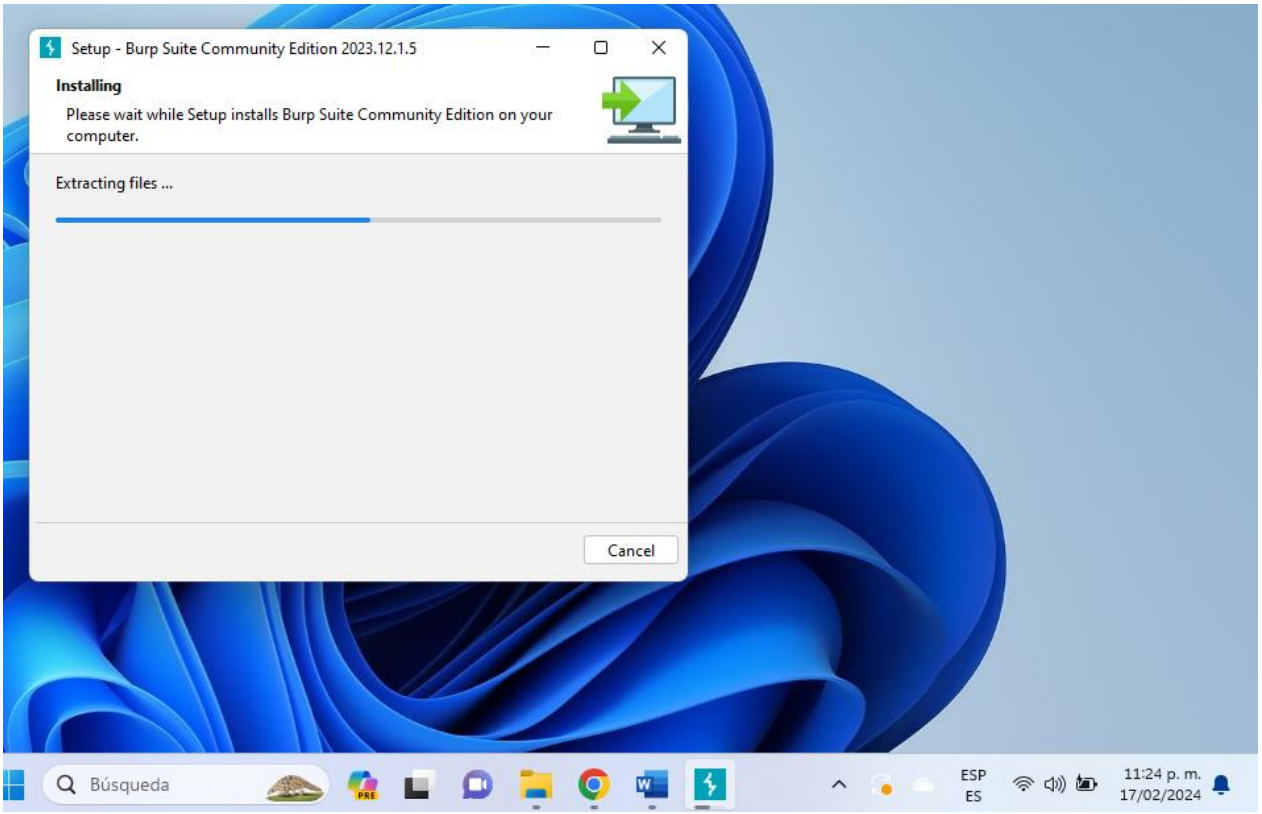
ello que esta vulnerabilidad también se conoce como inyección de objetos. Muchos ataques de deserialización finalizan antes de que se complete la deserialización. Esto significa que el proceso en sí mismo puede iniciar un ataque, sin importar si la funcionalidad de la aplicación interactúa o no directamente con el objeto malicioso. En definitiva, aún los sitios web y las aplicaciones basadas en lenguajes fuertemente tipados también pueden ser vulnerables a estas técnicas.

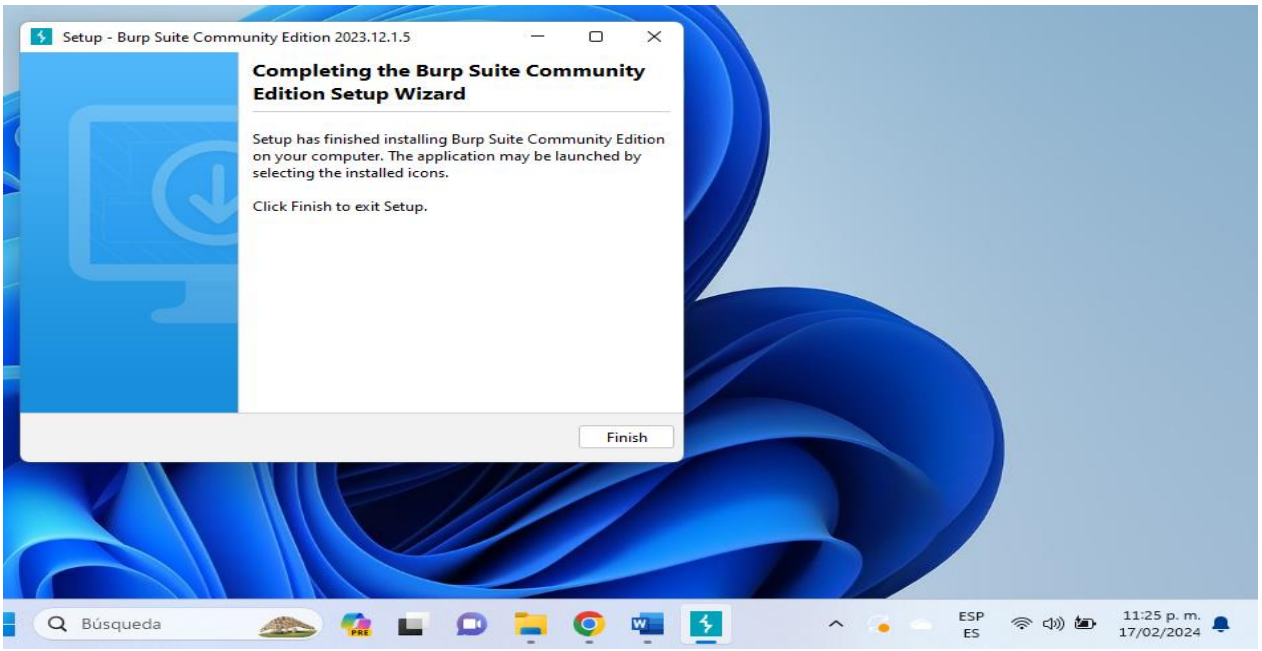
ATAQUE AL SITIO

En esta imágenes se muestra la instalación de burp suite community

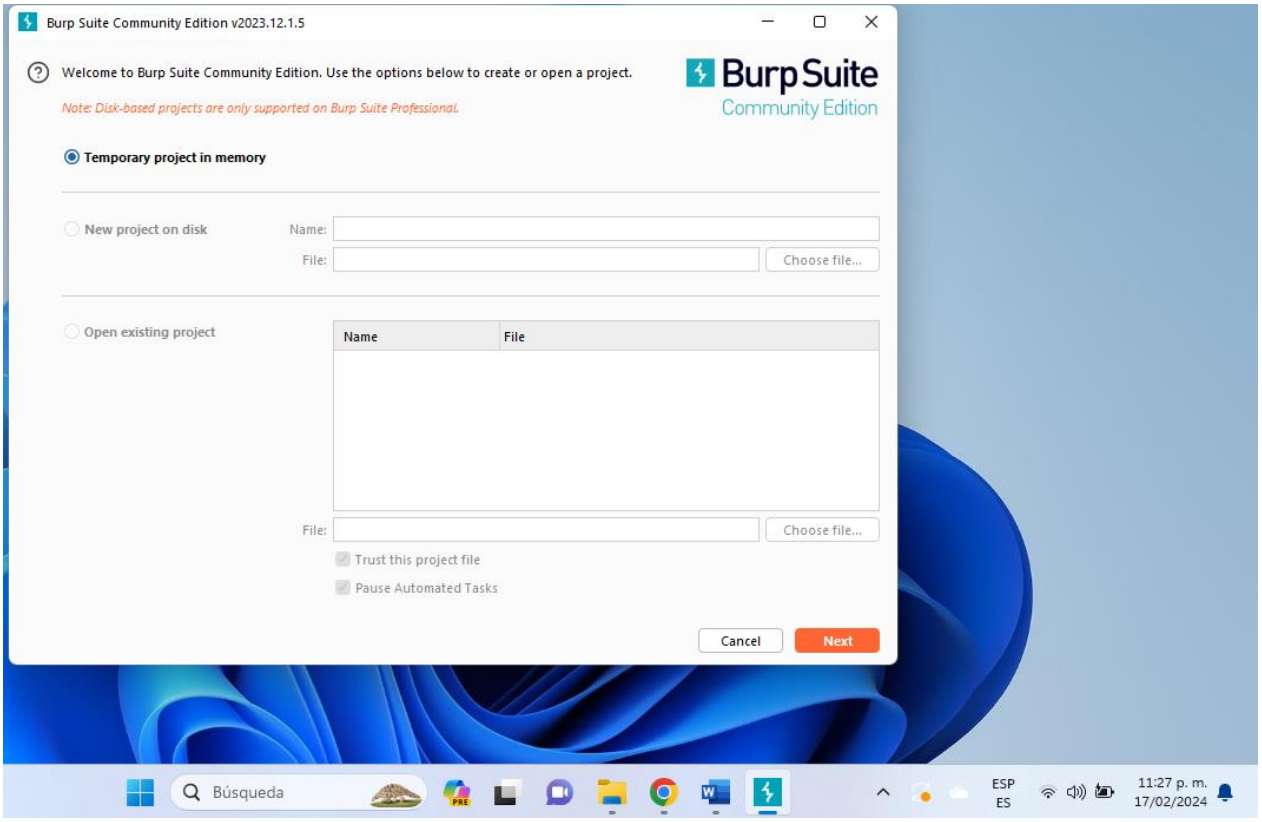



Setup - Burp Suite Community Edition 2023.12.1.5
Select Destination Directory
Where should Burp Suite Community Edition be installed?
Select the folder where you would like Burp Suite Community Edition to be installed, then click Next.
C:\Users\odio_\AppData\Local\Programs\BurpSuiteCommunity
Browse ...
Required disk space: 392 MB
Free disk space: 41 GB
< Back
Next >
Cancel
Búsqueda
ESP
ES
11:24 p. m.
17/02/2024

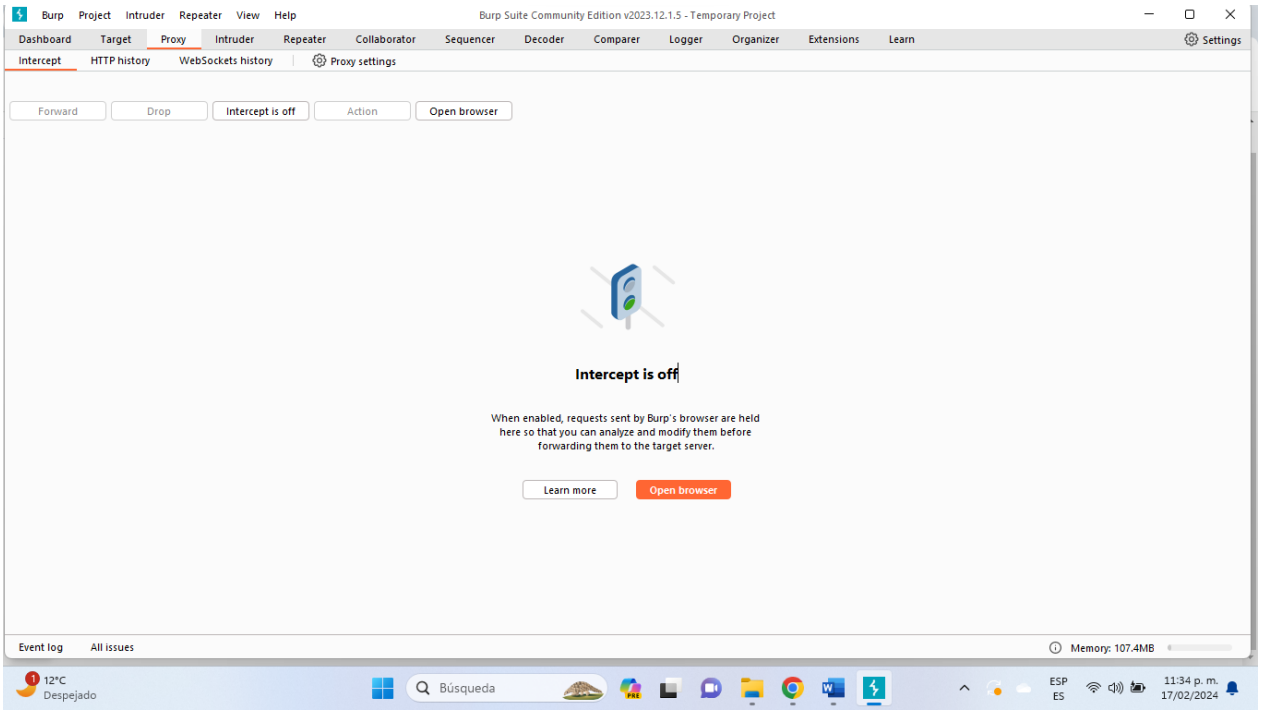

Setup - Burp Suite Community Edition 2023.12.1.5
Installing
Please wait while Setup installs Burp Suite Community Edition on your computer.
Extracting files ...
Cancel
Búsqueda
ESP
ES
11:24 p. m.
17/02/2024



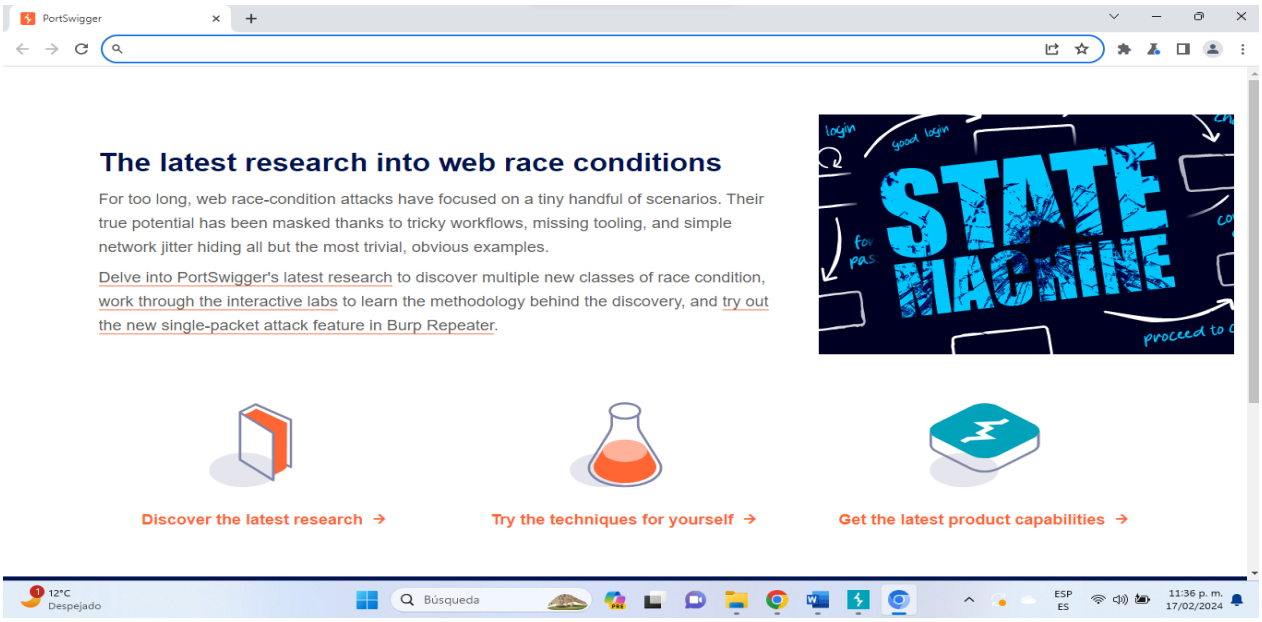
En esta imagen se muestra la entrada ala herramienta a utilizar



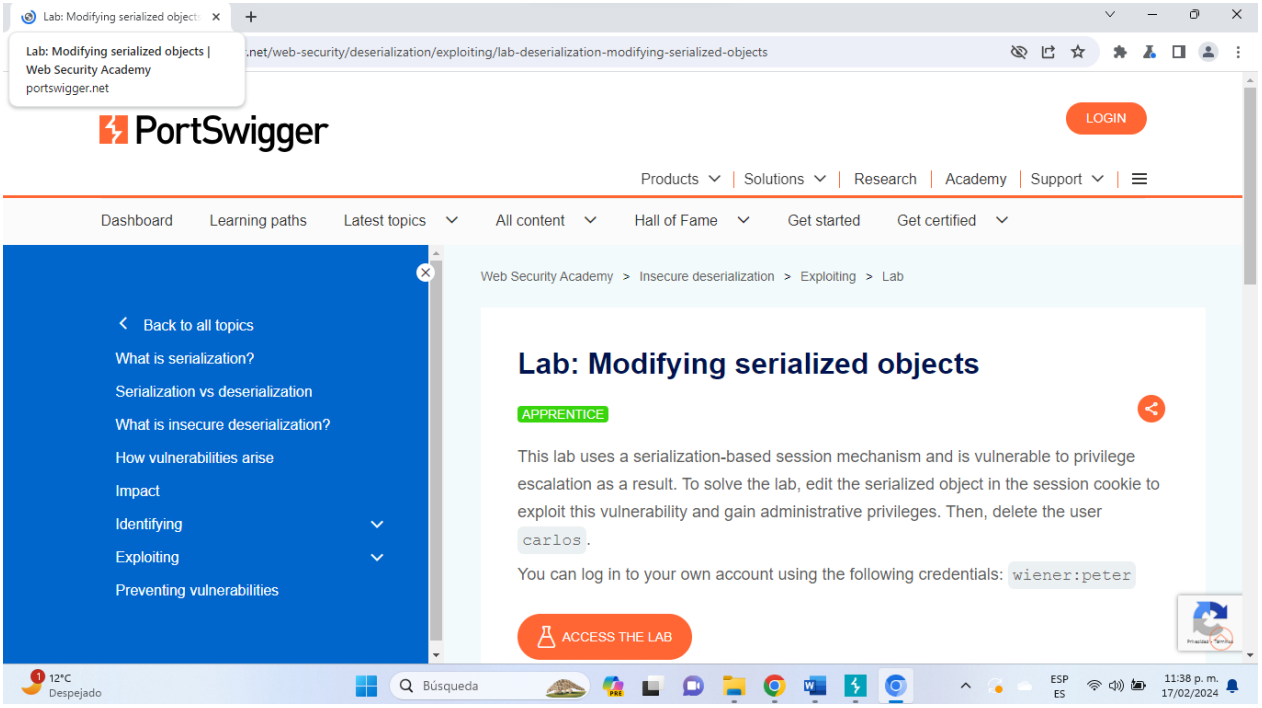
En esta imagen se muestra para ingresar ala pagina con open browser



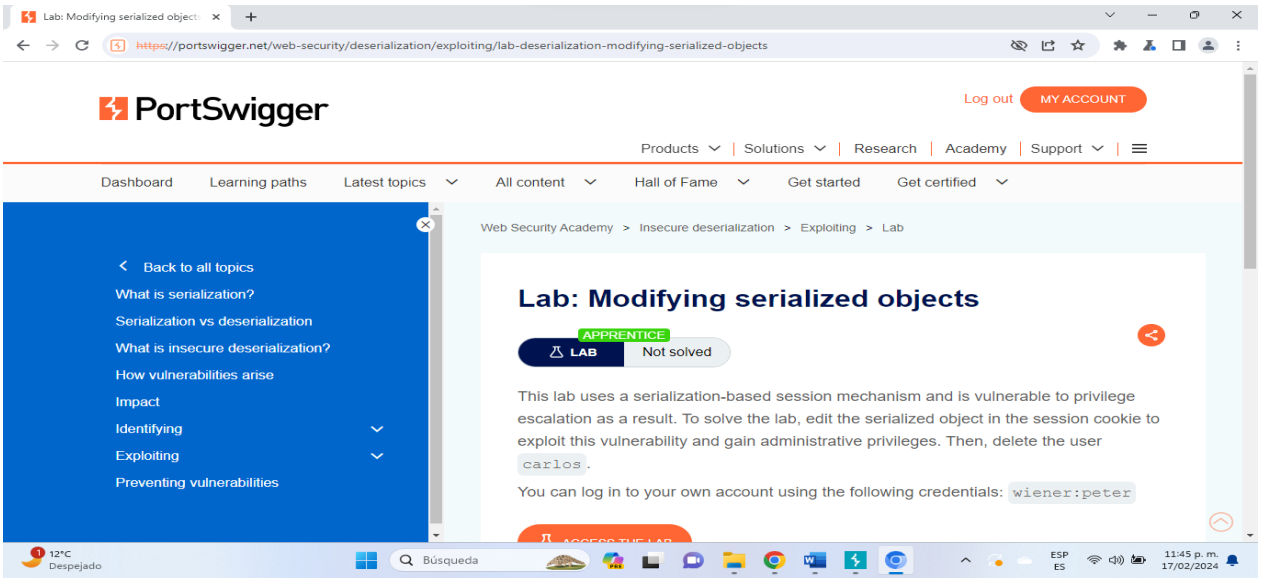
En esta imagen se muestra la entrada ala pagina donde se estara atrabajando



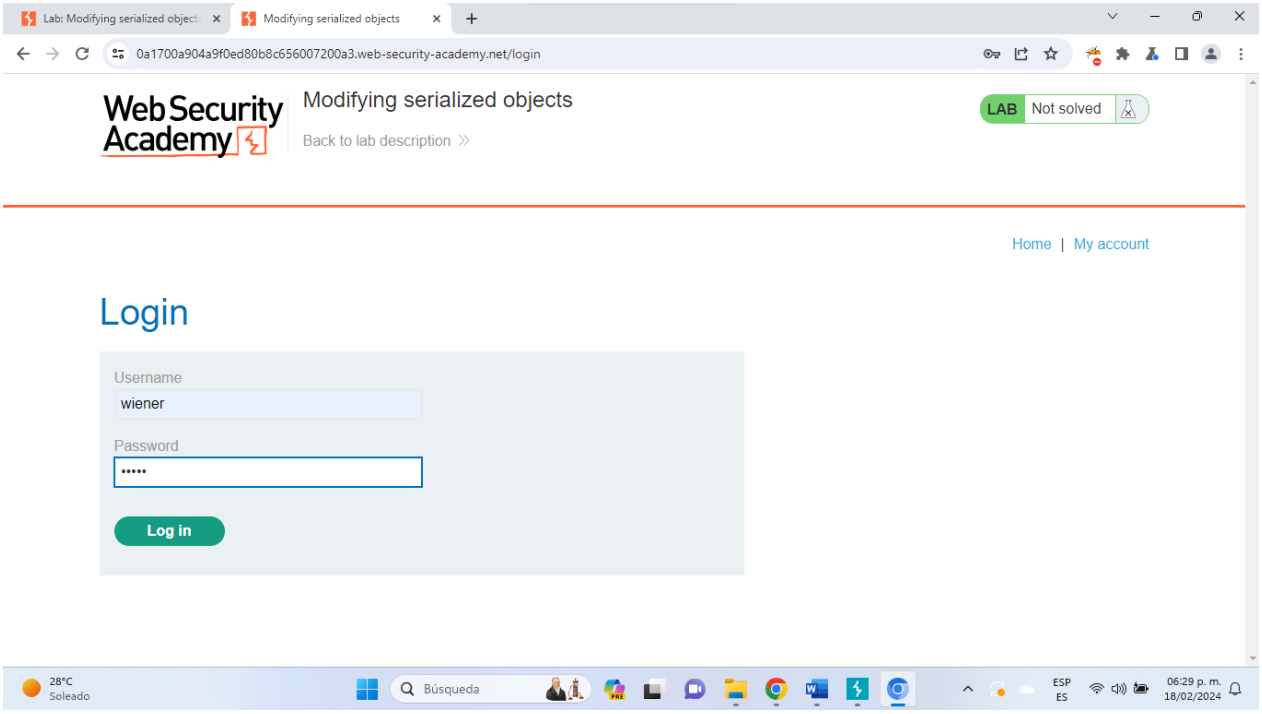
En esta imagen se muestra el enlace copiado para trabajar con el laboratorio

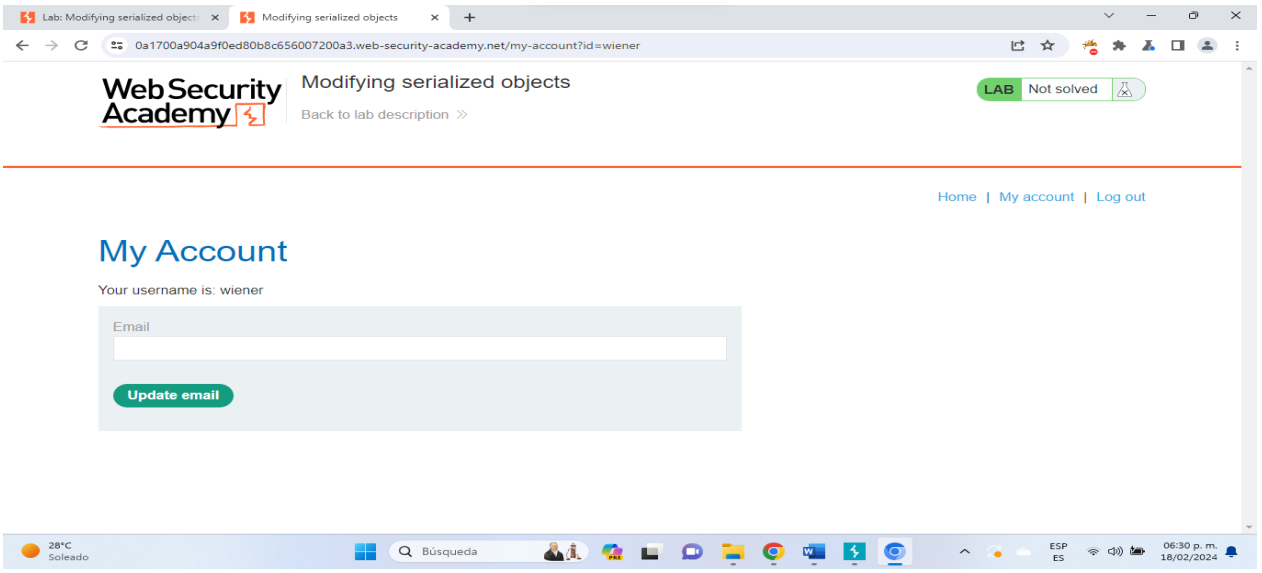


En esta imagen se muestra el incio de secion con el login creado



En estas imagenes se muestra el incio de secion en my account con el usuario: wiener contraseña: peter





CONCLUSION

En conclusión, en esta actividad se aprende que la deserialización insegura permite a un atacante manipular objetos serializados para pasar datos dañinos al código de la aplicación, e incluso, reemplazar un objeto serializado por un objeto de una clase distinta. Es decir, los objetos que estén disponibles para el sitio web serán deserializados e instanciados, independientemente de la clase que se esperaba. De hecho, es por ello que esta vulnerabilidad también se conoce como inyección de objetos. Muchos ataques de deserialización finalizan antes de que se complete la deserialización. Esto significa que el proceso en sí mismo puede iniciar un ataque, sin importar si la funcionalidad de la aplicación interactúa o no directamente con el objeto malicioso. En definitiva, aún los sitios web y las aplicaciones basadas en lenguajes fuertemente tipados también pueden ser vulnerables a estas técnicas.

También se aprende a utilizar la herramienta burp suite que es una herramienta ideal utilizada en el campo de la ciberseguridad, las pruebas de penetración y las auditorías. Es una herramienta muy utilizada en el hacking ético (la práctica que realizan especialistas en ciberseguridad con el objetivo de encontrar vulnerabilidades en un determinado sistema) debido a que analiza, evalúa, acepta o rechaza todas las solicitudes y respuestas de una aplicación determinada.

REFERENCIAS BIBLIOGRAFICAS

The WhiteHat Panther. (2022, 17 marzo). *Hackeando Login Page por Fuerza Bruta -PortSwigger Academy* [Vídeo]. YouTube. https://www.youtube.com/watch?v=NOYfWjsobZg

dokkillo. (2021, 8 diciembre). *Access Control Lab: funcionalidad de administrador desprotegida con url no predictible* [Vídeo]. YouTube. https://www.youtube.com/watch?v=lQhJ7I-3wJw

Castillo, A. (2020, 8 mayo). *Ataques en Sistemas de Autenticación – OWASP Top 2*. https://es.linkedin.com/pulse/ataques-en-sistemas-de-autenticaci%C3%B3n-owasp-top-2-alexander-castillo

Bnke0x0. (2022b, abril 5). *User ID Controlled by Request Parameter With Password Disclosure | PortSwigger (Video solution)* [Vídeo]. YouTube. https://www.youtube.com/watch?v=b2TXT8AYUqY