Ingeniería
en Desarrollo de Software

Actividad | 2 | Deserialización Insegura

Auditoría Informática

Ingeniería en Desarrollo de Software

TUTOR: Jessica Hernández Romero

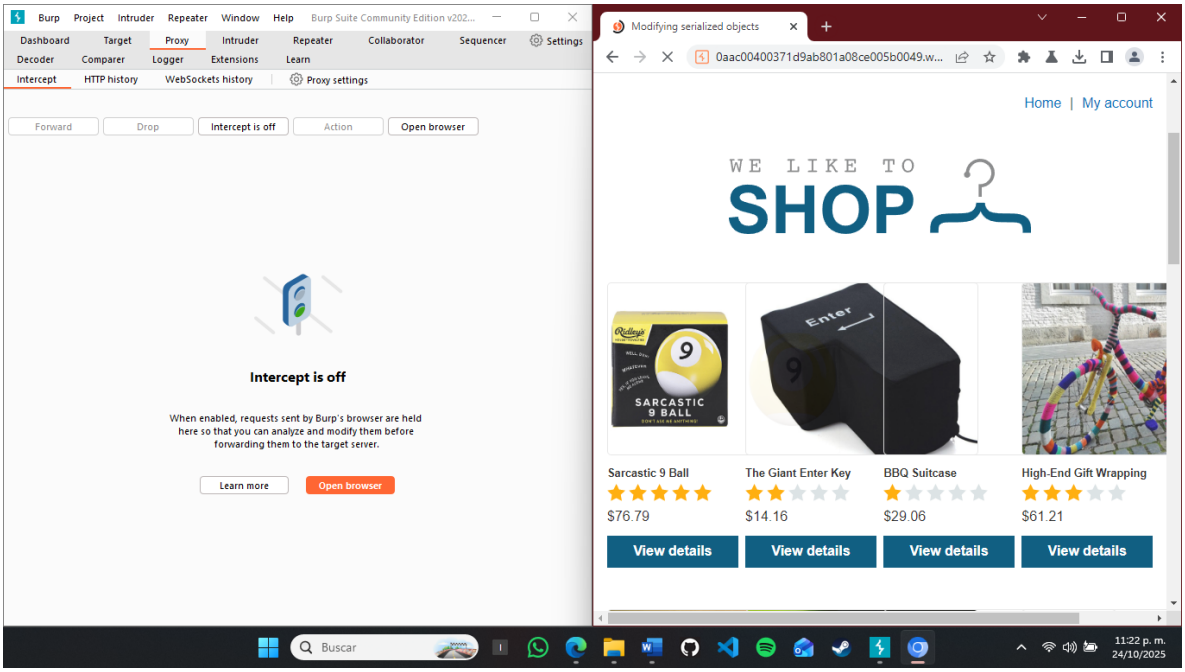
ALUMNO: Carlos Fco Estrada Salazar

FECHA: 25/ Oct /2025

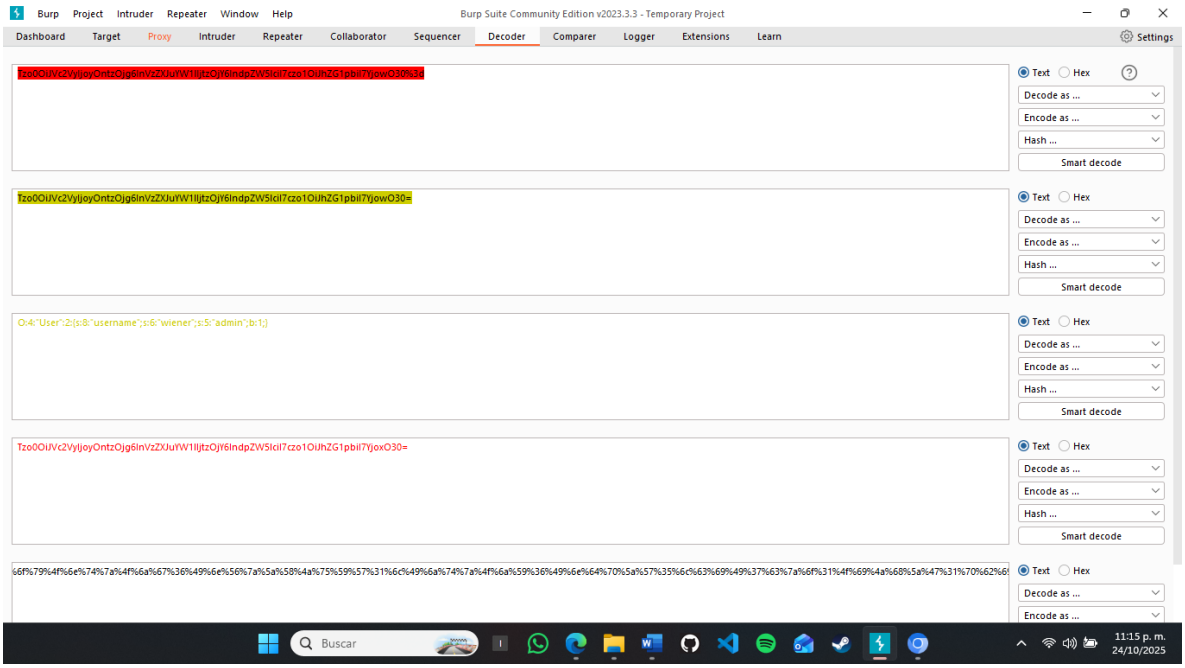
DESARROLLO

Ataque al sitio:

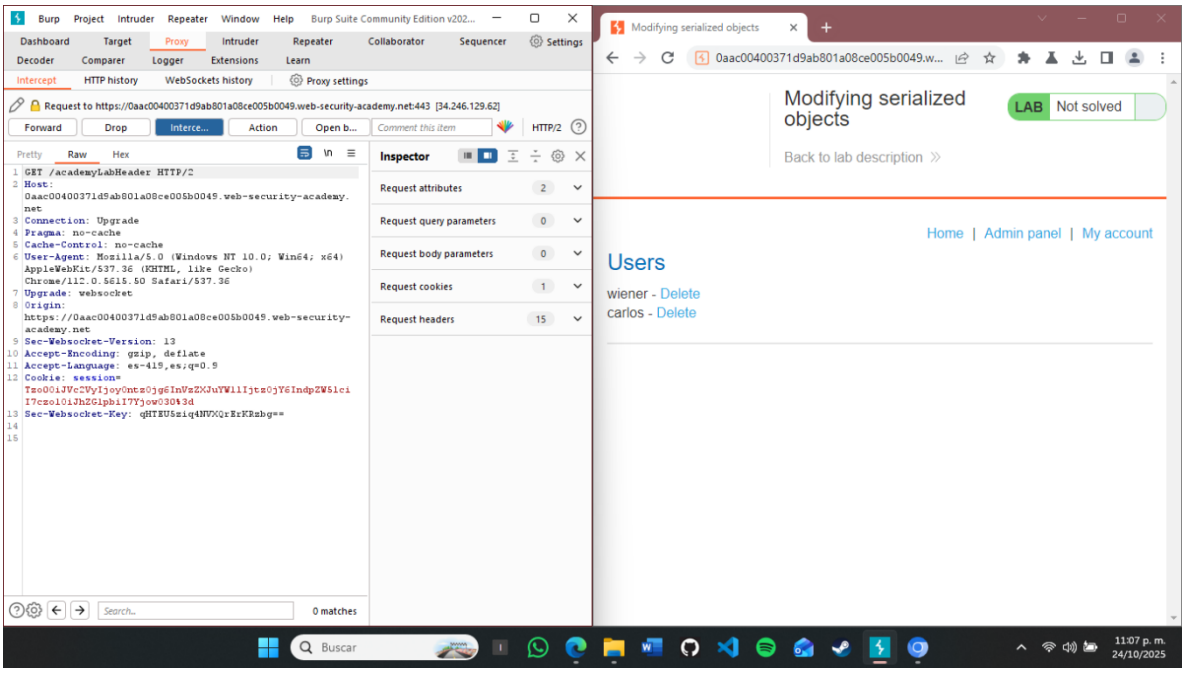
1. Se realiza la descarga e instalación de ***Burp Suite Community Edition***.
2. Entro a ***PortSwigger*** desde el navegador del sistema e inicio sesión, con la sesión abierta copeo el link de Lab.
3. Ejecuto el programa ***Burp Suite*** e ingrese a la sección de ***Proxy***, una vez hay ejecuto el navegador de ***Burp Suite***.



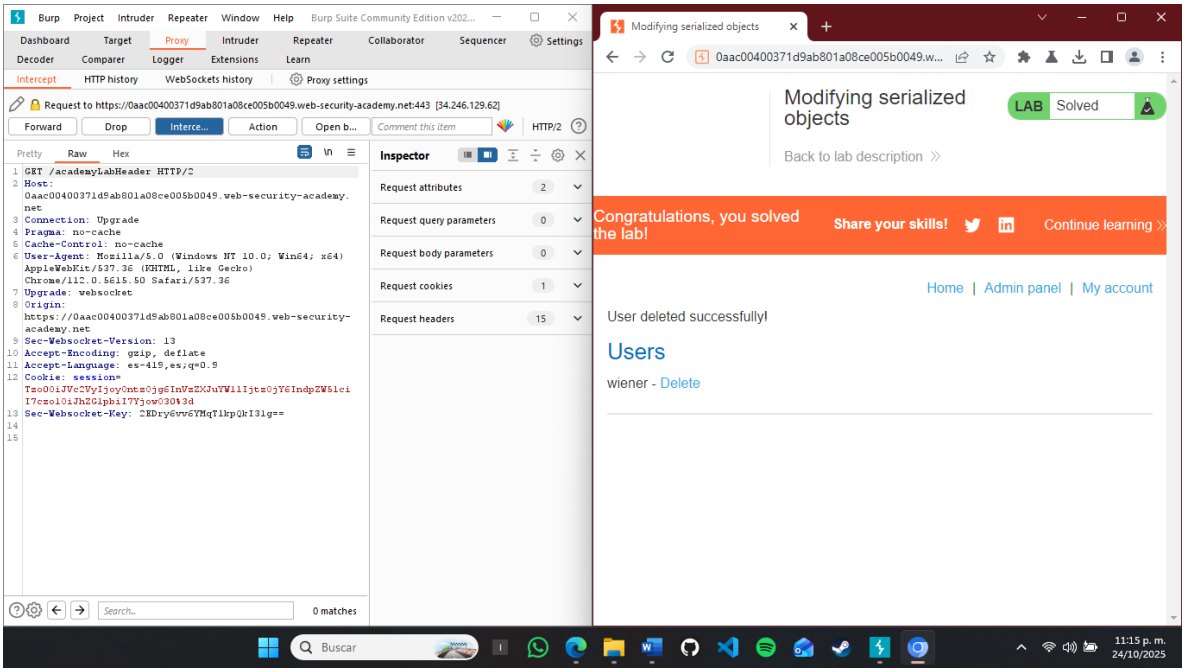
4. En la barra de búsqueda del navegador de ***Burp Suite*** pego el link que teníamos en el portapapeles y abro la página.
5. Ingreso a la opción ***My Acount*** e ingreso las credenciales proporcionadas ***wiener:Peter***.
6. Una vez echo lo anterior, en ***Burp Suite/Proxy*** ingreso a la opción ***HTTP history***, y busco la opción ***POST /login***, el cual me permitirá modificar la ***set-Cookie: sesión=***.
7. Envió la información de ***set-Cookie: sesión=*** al ***decodificador*** y realizo las acciones correspondientes.



8. ***Decodifico*** la información que encontré en ***set-Cookie: sesión=***, la primera vez como ***URL***, después a ***Base64***.
9. Al tener la cookie en ***Base64***, cambie el parámetro ***b: 0*** a ***b: 1***, esto para otorgar permisos de administrador al usuario.
10. Con el valor corregido, ***Codifico*** una vez mas a ***URL***, y copio la cookie resultante.
11. Con la cookie resultante en el portapapeles, regreso a ***Burp Suit/Proxy/Intercept/Intercept is off*** y recargo el navegador.
12. Con esto realizado, en el **Burp** aparecerá la información de usuario, se reemplaza la información de ***Cookie: sesión=*** por la que se tenía en el portapapeles y selecciono ***Forward***, con esto aparecerá la opción ***Admin panel*** en la interfaz del navegador y accedemos a la misma.
13. De regreso en ***Burp Suit/Proxy/Intercept/Raw*** se modifica la información de ***Cookie: sesión=*** por la que teníamos en el portapapeles y damos clic en ***Forward***.



14. De esta manera se accede al panel que solo los administradores pueden ver con los usuarios registrados.
15. Para eliminar al usuario Carlos damos clic en *carlos – Delete*.
16. Regresamos a ***Burp Suit/Proxy/Intercept/Raw*** y nuevamente remplazamos la cookie del usuario normal por la del administrador y damos clic en ***Forward.***



CONCLUSIÓN

La actividad de deserialización insegura realizada con ***Burp Suite*** y el laboratorio de ***PortSwigger*** expone de forma práctica riesgos reales que afectan tanto al ámbito laboral como a la vida cotidiana. En el entorno profesional, pone en evidencia que mecanismos de gestión de sesiones mal diseñados —por ejemplo, objetos serializados almacenados en cookies sin validación ni firma— pueden permitir la escalada de privilegios y el compromiso de cuentas administrativas. Esto tiene consecuencias directas: pérdida de integridad de datos, accesos no autorizados, interrupción de servicios y daños reputacionales que pueden traducirse en sanciones regulatorias y pérdidas económicas. Por ello, el ejercicio refuerza la necesidad de integrar auditorías de seguridad funcionales y automatizadas en el ciclo de vida del software (*SDLC*), aplicar principios de diseño seguro (validación, firma y encriptación de tokens), y capacitar a desarrolladores y equipos de operación en amenazas emergentes.

En la vida cotidiana, la misma vulnerabilidad explica por qué usuarios comunes pueden ver su información comprometida o sus cuentas manipuladas si las aplicaciones que usan carecen de controles adecuados. La práctica demuestra la importancia de exigir buenas prácticas por parte de proveedores (uso de *HTTPS*, tokens firmados, control de sesiones) y de adoptar medidas personales como contraseñas fuertes, autenticación multifactor y revisar permisos de aplicaciones.

Finalmente, el ejercicio fomenta una mentalidad crítica y ética: aprender a identificar, explotar (en entornos controlados) y mitigar vulnerabilidades prepara a los profesionales para diseñar sistemas más resilientes y a los usuarios para reconocer riesgos, reduciendo la superficie de ataque y fortaleciendo la confianza en los sistemas digitales.