Ejercicios Cookies

Ejercicio 1: Introducción a las Cookies HTTP-Only

Enunciado:
Investiga qué son las cookies HTTP-Only y responde a las siguientes preguntas:

1. ¿Qué diferencia hay entre una cookie normal y una cookie HTTP-Only?
2. ¿Por qué las cookies HTTP-Only son más seguras?
3. ¿Se pueden acceder a las cookies HTTP-Only desde JavaScript?

Requisitos:

1. Redacta tus respuestas en un documento.
2. Incluye ejemplos prácticos donde sea necesario.

Ejercicio 2: Configuración de Cookies HTTP-Only en el Servidor

Configura una cookie HTTP-Only en un servidor utilizando Node.js y Express. La cookie debe llamarse sessionToken y almacenar un valor ficticio.

Requisitos:

1. Utiliza el paquete cookie-parser en Express.
2. Establece la cookie con la opción httpOnly: true.
3. Verifica en el navegador que la cookie no sea accesible desde JavaScript.

Ejercicio 3: Verificación de Cookies HTTP-Only en el Navegador

Enunciado:
Después de configurar la cookie HTTP-Only en el servidor, abre las herramientas de desarrollo del navegador y responde:

1. ¿Dónde puedes ver la cookie HTTP-Only?
2. Intenta acceder a la cookie mediante document.cookie en la consola del navegador. ¿Qué resultado obtienes?

Requisitos:

1. Documenta los pasos seguidos y los resultados observados.
2. Explica por qué no puedes acceder a la cookie desde JavaScript.

Ejercicio 4: Simulación de Inicio de Sesión Seguro con Cookies HTTP-Only

Enunciado:
Crea una aplicación simple que permita a los usuarios iniciar sesión. Al iniciar sesión correctamente, el servidor debe enviar una cookie HTTP-Only llamada authToken. Esta cookie se debe utilizar para mantener la sesión del usuario.

Requisitos:

1. Configura el servidor para enviar la cookie authToken al iniciar sesión.
2. Protege una ruta /perfil que solo sea accesible si la cookie está presente.
3. Implementa una opción para cerrar sesión eliminando la cookie.

Ejercicio 5: Comparación entre Cookies HTTP-Only y localStorage

Enunciado:
Investiga las diferencias entre almacenar un token de autenticación en una cookie HTTP-Only y en localStorage. Responde a las siguientes preguntas:

1. ¿Cuál es más vulnerable a ataques XSS (Cross-Site Scripting)?
2. ¿Qué método es más seguro para almacenar tokens sensibles?
3. ¿Qué ventajas y desventajas tiene cada uno?

Requisitos:

1. Redacta tus respuestas con ejemplos prácticos.
2. Incluye recomendaciones sobre cuándo usar cada método.

Ejercicio 6: Simulación de un Ataque XSS y la Protección con Cookies HTTP-Only

Enunciado:
Crea una página web vulnerable a XSS que permita a los usuarios ingresar comentarios. Luego, intenta inyectar un script malicioso que intente robar cookies mediante document.cookie.

1. Primero, almacena un token en localStorage y observa si el ataque puede acceder al token.
2. Luego, almacena el mismo token en una cookie HTTP-Only y verifica si el ataque puede acceder a la cookie.

Requisitos:

1. Documenta el resultado del ataque en ambos casos.
2. Explica cómo las cookies HTTP-Only previenen el robo de datos sensibles.

Ejercicio 7: Configuración de Cookies Seguras con HTTPS y HTTP-Only

Enunciado:
Modifica tu servidor para que las cookies HTTP-Only también sean seguras, utilizando la opción secure: true. Asegúrate de que la cookie solo se envíe si la conexión es HTTPS.

Requisitos:

1. Configura la cookie con httpOnly: true y secure: true.
2. Prueba la aplicación en un entorno HTTPS y verifica que la cookie no se envíe a través de HTTP.
3. Documenta tus hallazgos.

Ejercicio 8: Expiración y Renovación de Cookies HTTP-Only

Enunciado:
Configura una cookie HTTP-Only con una duración limitada de 5 minutos (maxAge). Implementa una funcionalidad que renueve la cookie automáticamente si el usuario está activo en la aplicación.

Requisitos:

1. Configura la expiración de la cookie en el servidor.
2. Implementa una función que detecte la actividad del usuario (por ejemplo, movimientos del ratón o clics) y renueve la cookie antes de que expire.
3. Documenta cómo funciona la renovación de la cookie.

Ejercicio 9: Protección CSRF con Cookies HTTP-Only y Tokens

Enunciado:
Crea una aplicación que utilice cookies HTTP-Only para la autenticación. Implementa un token CSRF que se envíe con cada solicitud desde el cliente y se verifique en el servidor.

Requisitos:

1. Genera un token CSRF en el servidor y envíalo como parte de la respuesta.
2. Incluye el token CSRF en las solicitudes POST desde el cliente.
3. Verifica el token CSRF en el servidor para validar las solicitudes.

Ejercicio 10: Visualización y Gestión de Cookies en el Navegador

Enunciado:
Después de haber configurado cookies HTTP-Only en el servidor, realiza las siguientes tareas en el navegador:

1. Accede a la pestaña de "Application" en las herramientas de desarrollo y localiza las cookies configuradas.

2. Intenta modificar o eliminar una cookie HTTP-Only desde las herramientas del navegador. ¿Es posible?
3. Documenta los resultados y explica por qué las cookies HTTP-Only no se pueden modificar desde el cliente.