Tarea online PPS03.

Título de la tarea: Pruebas de XSS

Unidad: 3

Ciclo formativo y módulo: Curso de especialización en ciberseguridad en entornos de las tecnologías de la información - Puesta en Producción Segura.

¿Qué contenidos o resultados de aprendizaje trabajaremos?

Resultados de aprendizaje

- **RA3.** Detecta y corrige vulnerabilidades de aplicaciones web analizando su código fuente y configurando servidores web.

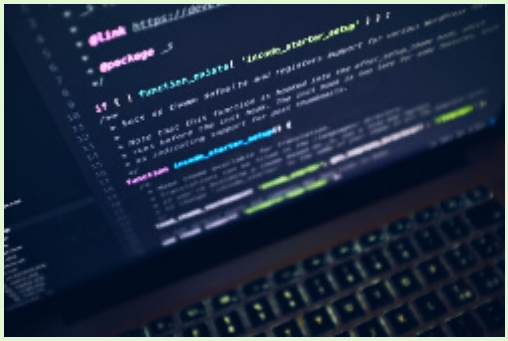
Contenidos

Detección y corrección de vulnerabilidades de aplicaciones web.

1. Desarrollo seguro de aplicaciones web.
2. Entrada basada en formularios.
3. Estándares de autenticación y autorización.
4. Robo de sesión. Ataque por fuerza bruta. Sniffing. Propagación en URL. Servidores compartidos. Métodos de prevención
5. Vulnerabilidades web.
6. Almacenamiento seguro de contraseñas.
7. Contramedidas.
8. Seguridad de portales y aplicativos web

1.- Descripción de la tarea.

Caso práctico



@lucabravo para unsplash, *computadora-portatil-gris-encendida* (Licencia Unsplash)

Julián ha estado probando distintos tipos de vulnerabilidades y ataques que podría sufrir su aplicativo web, pero aún tiene pruebas y escenarios que testar.

Sus compañeros de trabajo le han comentado que vigile si su aplicativo web pudiera ser víctima de algún tipo de *Cross Site Scripting (XSS).*

Para ello, **Julián** revisará un formulario para introducir comentarios en un foro que le preocupa pueda ser víctima de algún tipo de XSS.

¿Qué te pedimos que hagas?

- **Apartado 1: Inyección de *Cross Site Scripting* (XSS)**

1.- Trabajaremos sobre el entorno de DVWA que construimos en la unidad anterior.
2.- Arranca el aplicativo de DVWA (en la unidad 2 ya vimos como lanzarla) y accede en un navegador a **http://localhost:4280**
3.- Nos desplazaremos hasta el módulo de XSS del menú lateral izquierdo. En este caso de tipo **XSS almacenado (XSS Stored)**
4.- Seguimos trabajando en el modo **Low** (configurado en el menú de *Security*)
5.- ¿Podrías conseguir que muestre una ventana de alerta con un mensaje cada vez que alguien visita el libro de firmas?
Ayuda: los navegadores interpretan Javascript. Si los mensajes que se graban en esta pantalla se muestran a todos los usuarios, y grabo como mensaje un código javascript, este se podría ejecutar en todos los clientes que abran esa página si la aplicación no está protegida.
6.- ¿Serías capaz de introducir un XSS que redirija al usuario a una web de tu elección cada vez que se visite el libro de firmas? (por ejemplo Youtube.com o Google.com)
7.- ¿Serías capaz de robar la cookie del usuario? ¿Por ejemplo redirigiéndola a un servidor tuyo?
Ayuda: puedes crear de forma sencilla un servidor web que escuche peticiones en tu máquina local mediante un contenedor (por ejemplo mira

https://hub.docker.com/r/trinitronx/python-simplehttpserver). Ejecuta la siguiente línea de comandos:

docker run --name simplehttp --rm -p 8080:8080 trinitronx/python-simplehttpserver

Con esto podrás ver en la consola/terminal como recibe tu servidor la cookie

NOTA: para terminar un contenedor lanzado en primero plano se puede simplemente cerrar la terminal. Después para terminar el contenedor ejecuta:

docker rm -f simplehttp

- **Apartado 2: Preguntas sobre el ejercicio**

1.- ¿Qué tipo de ataques son los XSS? ¿Qué diferencia vemos respecto a los ataques de tipo SQL Inyection?
2.- ¿Podrías decir en dónde se inyectan (guardan) los comandos escritos en el formulario del apartado 1?
3.- ¿Podrías decir si en el caso del apartado 1 este XSS es temporal o permanente? Justifica la respuesta
4.- ¿Qué mejoras ves que se podrían hacer tanto en lado cliente como en el lado servidor comentadas durante el curso?¿Se podría evitar XSS sólo en el lado del navegador? Justifica la respuesta
5.- ¿Qué controles de seguridad tiene implementados en el modo **Low**? ¿y en el modo **Impossible**?
6.- ¿Por qué es un riesgo este formulario tal cual está?

NOTA IMPORTANTE

Para todos los apartados es necesario entregar las capturas de pantalla de los principales pasos realizados, explicando el proceso seguido en cada uno de ellos. Las capturas de pantalla realizadas deben tener como fondo de pantalla la plataforma con tu usuario mostrando claramente la foto de tu perfil. Aquellos apartados/subapartados que no cumplan esta condición no serán corregidos.

2.- Información de interés.

Recursos necesarios y recomendaciones

Recursos necesarios

- Ordenador personal con, al menos, 4 Gigabytes de memoria RAM.
- Conexión a Internet.
- Sistemas Operativos Windows 10, Ubuntu 20.04.
- Docker.
- Navegador web.

Recomendaciones

- Antes de abordar la tarea:
  - Lee con detenimiento la unidad, consulta los enlaces para saber más, examina el material proporcionado por el profesor y aclara las dudas que te surjan con él.
  - Realiza el examen online de la unidad, y consulta nuevamente las dudas que te surjan. Solo cuando lo tengas todo claro, debes abordar la realización de la tarea.
- No olvides elaborar el documento explicativo.
- Para entender mejor los ataques de tipo inyección de XSS puedes visitar los siguientes enlaces:
  - https://owasp.org/www-community/attacks/xss/
  - https://es.wikipedia.org/wiki/Cross-site_scripting
- Tienes una guía de comandos de XSS disponible en https://portswigger.net/web-security/cross-site-scripting/cheat-sheet
- Si tienes problemas con la longitud del campo de texto del formulario puedes darle botón derecho sobre el cuadro de texto, luego a Inspeccionar y cambiar la longitud (variable maxlength por defecto está en 50)

Indicaciones de entrega

Una vez realizada la tarea, el envío se realizará a través de la plataforma. El archivo se nombrará siguiendo las siguientes pautas:

Apellido1_Apellido2_Nombre_PPS_Tarea03

3.- Evaluación de la tarea.

Criterios de evaluación implicados

Criterios de evaluación RA3

- a. Se han validado las entradas de los usuarios.
- b. Se han detectado riesgos de inyección tanto en el servidor como en el cliente.
- c. Se ha gestionado correctamente la sesión del usuario durante el uso de la aplicación.
- d. Se ha hecho uso de roles para el control de acceso.
- e. Se han utilizado algoritmos criptográficos seguros para almacenar las contraseñas de usuario.
- f. Se han configurado servidores web para reducir el riesgo de sufrir ataques conocidos.
- g. Se han incorporado medidas para evitar los ataques a contraseñas, envío masivo de mensajes o registros de usuarios a través de programas automáticos (bots).

¿Cómo valoramos y puntuamos tu tarea?

Rúbrica de la tarea	
Apartado 1.5: ¿Podrías conseguir que muestre un mensaje cada vez que alguien visita el libro de firmas?	1 punto
Apartado 1.6: ¿Serías capaz de introducir un XSS que redirija al usuario a una web de tu elección cada vez que se visite el libro de firmas? (por ejemplo Youtube.com o Google.com)	1 punto
Apartado 1.7: ¿Serías capaz de robar la cookie del usuario? ¿Por ejemplo redirigiéndola a un servidor tuyo?	2 puntos
Apartado 2.1: ¿Qué tipo de ataques son los XSS? ¿Qué diferencia vemos respecto a los ataques de tipo SQL Inyection?	1 punto
Apartado 2.2:¿Podrías decir en dónde se inyectan (guardan) los comandos escritos en el formulario del apartado 1?	1 punto

Apartado 2.3: ¿Podrías decir si en el caso del apartado 1 el XSS es temporal o permanente? Justifica la respuesta	1 punto
Apartado 2.4: ¿Qué mejoras ves que se podrían hacer tanto en lado cliente como en el lado servidor comentadas durante el curso?¿Se podría evitar XSS sólo en el lado del navegador? Justifica la respuesta	1 punto
Apartado 2.5: ¿Qué controles de seguridad tiene implementados en el modo Low?¿Y en el modo Impossible?	1 punto
Apartado 2.6:¿Por qué es un riesgo este formulario tal cual está?	1 punto

NOTA IMPORTANTE

Aquellos apartados/subapartados en los que las capturas de pantalla no sean claras o no tengan como fondo de pantalla la plataforma con tu usuario mostrando claramente la foto de tu perfil, no serán corregidos.