Apuntes SinCara Extras XSS, CSRF y SSRF

XSS - Cross-Site Scripting - Secuencia de Comandos en Sitios Cruzados

- https://cwe.mitre.org/data/definitions/79.html - CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting'). Número 1 del top 25:
  - https://cwe.mitre.org/top25/index.html
- A una URL legítima y válida, se le añade un script o código que se ejecuta en el cliente (Tipo 1), o también puede estar añadido en un comentario de esa página (Tipo 2).
- Es una URL especialmente construida, normalmente en ataques de Phishing.
- Los XSS ocurren cuando:
  - una aplicación toma datos no confiables y los envía al navegador web sin una validación y codificación apropiada;
  - o actualiza una página web existente con datos suministrados por algún usuario utilizando una API que ejecuta JavaScript en el navegador.
- Existen tres formas usuales de XSS para atacar a los navegadores de los usuarios
  - **Tipo 1: XSS Reflejado o No Persistente**: Este ataque no se ejecuta en la aplicación web, sino que es una inyección de código que se inicia en el momento que el usuario pincha en un enlace especialmente construido que contiene el código malicioso.
    - http://example.com/search?query=<script>alert(1)</script>
  - **Tipo 2: XSS Almacenado o Persistente**: En esta modalidad se busca inyectar código en el sitio web, lo que lo hace más peligroso debido a que no afecta a un solo usuario sino a todas las personas que visiten el sitio web. Estos tipos de ataques se encuentran principalmente en sitios que permiten alguna clase de entrada de datos sin validar, como por ejemplo blogs, formularios, foros, tweets...
  - **Tipo 0: XSS Basados en DOM, o XSS local** (Document Object Model-based): Es una variante del Tipo 1 o Reflejado. Tiene una diferencia esencial con los otros dos tipos, y es que el código malicioso se inyecta localmente desde una URL externa o un código JavaScript, pero gracias al # no se carga desde la web, no se retransmite al servidor, se ejecuta solo en local. En este caso el daño se provoca por medio de los scripts que están en el lado del cliente, y no existen en el lado del servidor. (NOTA: la # se utiliza para indicar en que parte de la web queremos posicionarnos al cargarla, pero esa información es para el navegador del cliente, esa parte no se envia al servidor)
    - http://example.com/index.html#default=Payload
- Los ataques XSS incluyen el robo de la sesión, apropiación de la cuenta, evasión de autentificación de múltiples pasos, reemplazo de nodos DOM, inclusión de troyanos de autentificación, ataques contra el navegador, defacements, descarga de software malicioso, keyloggers, y otros tipos de ataques del lado cliente.
- Variantes:
  - **HTTP Response Splitting Attack** - En vez de añadir código, lo que se añade es otra cabecera HTTP, de tal forma que se hacen dos peticiones simultáneas, la original y legítima y visible, y la oculta y maliciosa, sin conocimiento del que hace la petición. Se puede juntar con CSRF y otros ataques.
    - https://www.cyberseguridad.net/index.php/247-crlf-injection-http-response-splitting - HTTP Responses Splitting Attack
      - http://www.abc.com/page.php?page=%0d%0aContent-Type:text/html%0d%0aHTTP/1.1 200 OK%0d%0aContent-Type: text/html%0d%0a%0d%0a%3Chtml%3EHacker%3C/html%3E
  - https://www.acunetix.com/blog/articles/universal-cross-site-scripting-uxss/ - **UXSS** is a type of attack that exploits client-side vulnerabilities in the browser or browser extensions in order to generate an XSS condition, and execute malicious code.
- Enlaces:
  - https://xss-game.appspot.com/ - Juego XSS

    - https://d3adend.org/xss/ghettoBypass - Ejemplos de ataques XSS
    - https://betanews.com/2005/10/13/cross-site-scripting-worm-hits-myspace/ - Ataque XSS en MySpace
    - https://netsec.expert/posts/xss-in-2021/ - XSS in 2021

CSRF - Cross-Site Request Forgery - Falsificación de Petición en Sitios Cruzados

- https://cwe.mitre.org/data/definitions/352.html - CWE-352: Cross-Site Request Forgery (CSRF). Número 4 del top 25:
    - https://cwe.mitre.org/top25/index.html
- Esta vulnerabilidad consiste en conseguir que el usuario haga una petición, sin su conocimiento, a una página web en la que esté previamente autenticado.
- Se basa generalmente en que los usuario suelen tener el «estar logueado permanentemente» activo, que por seguridad, y concretamente para evitar este ataque, no deberíamos activar.
- Cómo esa petición se lanza desde el navegador del cliente, utiliza sus cookies para realizarla, sin que se entere.
- Una posibilidad es a través de una URL especialmente creada, como:
    - https://web.com/cambiarPass.php?pass=1234&confirmarPass=1234 - Para cambiar la pass del usuario sin que se entere
- Aunque también puede darse el caso de apoyarse en una web especialmente creada por el ciberdelincuente, en la cual se embebe una petición oculta a una página conocida (facebook, etc..), por ejemplo a través de iframes, generalmente invisibles.
- https://blog.evidaliahost.com/cross-site-request-forgery-csrf/ - ¿Qué es el Cross-site request forgery o CSRF?

SSRF - Server Side Request Forgery - Falsificación de Solicitudes del Lado del Servidor

- https://cwe.mitre.org/data/definitions/918.html - CWE-918: Server-Side Request Forgery (SSRF). Número 19 del top 25:
    - https://cwe.mitre.org/top25/index.html
- Los ataques de tipo **SSRF** se producen en aplicaciones web inseguras que permiten a un atacante forzar **al servidor web** a realizar peticiones desde dentro del sistema, ya sea hacia el exterior o para exfiltrar datos o credenciales.
- Este tipo de vulnerabilidades se da cuando un atacante tiene la habilidad de hacer que el servidor objetivo inicie una nueva conexión contra otro equipo, independientemente de si al equipo remoto al que se conecta se encuentra dentro o fuera de su red interna.
- La principal ventaja para un atacante de que las peticiones sean realizadas desde dentro de la red en la que se encuentra el sistema vulnerable es que le van a permitir acceder a sitios que de otra manera no podría (**pivoting**).
    - https://www.elladodelmal.com/2015/04/ssrf-server-side-request-forgery-xspa.html
- https://www.blog.binaria.uno/2019/12/30/diferencia-entre-falsificacion-de-solicitudes-entre-sitios-y-del-lado-del-servidor/ - Diferencias entre CSRF y SSRF.
    - CSRF está orientado al cliente, SSRF al servidor.