📘 Glosario: Explotación de Vulnerabilidades Comunes con Python

---

1. SQL Injection (SQLi)

Vulnerabilidad que permite insertar o manipular instrucciones SQL a través de entradas del usuario mal validadas. Puede provocar acceso no autorizado, fuga de datos, modificación de bases de datos o incluso control total del sistema gestor. Es una de las vulnerabilidades más críticas del OWASP Top 10.

---

2. Cross-Site Scripting (XSS)

Tipo de vulnerabilidad que permite la inyección de scripts maliciosos en contenido visualizado por otros usuarios. Puede ser utilizado para robar cookies, secuestrar sesiones, registrar teclas o realizar ataques de phishing. Se clasifica en reflejado, almacenado y basado en DOM.

---

3. Payload

Fragmento de código o instrucción especialmente diseñado para provocar una respuesta específica en el sistema objetivo. En el contexto de SQLi y XSS, un payload puede ser una consulta maliciosa o un script HTML/JavaScript inyectado.

---

4. OWASP Top 10

Lista mantenida por la Open Web Application Security Project que agrupa las diez amenazas más críticas para aplicaciones web. Sirve como referencia para desarrolladores, auditores y equipos de ciberseguridad a nivel global.

---

5. Requests (Python)

Librería de Python que facilita el envío de solicitudes HTTP. Es utilizada ampliamente en pruebas de penetración automatizadas para interactuar con aplicaciones web, APIs REST y endpoints vulnerables.

---

6. BeautifulSoup

Librería de Python para el análisis y navegación de documentos HTML o XML. Permite localizar formularios, etiquetas y atributos, lo que facilita la automatización de tareas como detección de campos vulnerables o reflejo de entradas.

---

7. Fuzzing

Técnica de análisis dinámico que consiste en enviar datos inesperados o aleatorios a un sistema con el objetivo de encontrar fallos, excepciones o comportamientos anómalos. Se utiliza en el descubrimiento de nuevas vulnerabilidades.

---

8. Enumeración de Parámetros

Proceso de descubrimiento y prueba sistemática de los campos de entrada de una aplicación, como formularios o parámetros en URLs. Es una etapa clave en la detección de vectores vulnerables a ataques como XSS o SQLi.

---

9. Validación del Lado del Servidor

Conjunto de medidas implementadas en el backend para controlar, sanear y limitar las entradas del usuario antes de que se procesen. Es esencial para prevenir ataques por inyección, incluso cuando existe validación en el cliente.

---

10. Automatización de Pentesting

Uso de scripts y herramientas programadas para realizar tareas repetitivas durante un análisis de seguridad: escaneo, explotación, recolección de evidencias, generación de reportes. Python es uno de los lenguajes más utilizados para este fin.

---

11. Consentimiento Informado (en Pentesting)

Permiso explícito, documentado y delimitado que el analista debe recibir antes de realizar cualquier tipo de prueba ofensiva sobre sistemas ajenos. Su ausencia puede implicar consecuencias legales severas, incluso si no se causan daños.

---

12. Reflected XSS

Tipo de Cross-Site Scripting en el que el payload malicioso es inmediatamente reflejado en la respuesta del servidor. Suele ser explotado mediante enlaces especialmente diseñados que inducen al usuario a ejecutar código malicioso.

---

13. Stored XSS

Tipo de XSS donde el código malicioso es almacenado en el servidor (por ejemplo, en una base de datos) y ejecutado cada vez que otro usuario accede al contenido afectado. Es especialmente peligroso por su carácter persistente.

---

14. SQLMap

Herramienta de código abierto especializada en la automatización de ataques de inyección SQL. Soporta múltiples motores de bases de datos, técnicas de evasión, escalada de privilegios y extracción de datos sensibles.

---

15. Selenium

Framework de automatización para navegadores. En el contexto de seguridad, se utiliza para probar comportamientos dinámicos del cliente y verificar si scripts maliciosos (como en XSS) son ejecutados en tiempo real.
