[[](https://platzi.com/clases/pentesting/)Curso de Fundamentos de Pentesting](https://platzi.com/clases/pentesting/)

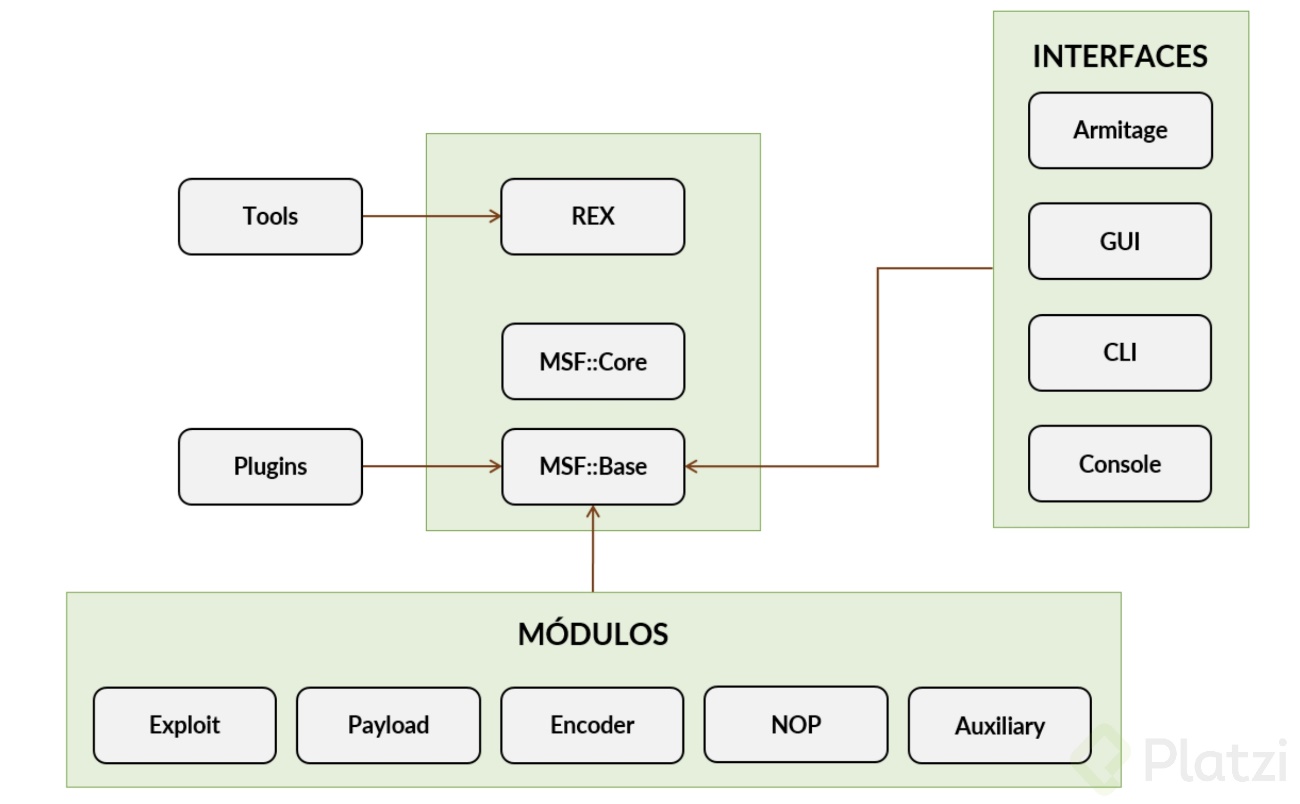
**ArtículoArquitectura de Metasploit**

[**Juan Pablo Caro**](https://platzi.com/@juanpcaro)

12 de Agosto de 2019

Para entender claramente cómo se comporta internamente Metasploit, y cómo se almacenan e interconectan los diferentes elementos que conforman su base de datos, es necesario analizar la arquitectura de framework en la que está soportado.

De manera general, podemos visualizar la arquitectura de Metasploit así:



Vamos a analizar por partes este diagrama:

**Núcleo de Metasploit:**

El núcleo de Metasploit está formado por 3 módulos o componentes:

1. **Módulo “Core”**: El módulo core es la API básica de Metasploit, contiene el proceso principal de ejecución y la definición general del programa.
2. **Módulo Base:** El módulo base es una API de segundo nivel. Proporciona recursos para que los diferentes módulos, herramientas y paquetes se conecten y puedan acceder a los procesos del Core.
3. **Librería REX:** La librería REX es la que realiza las operaciones entre los módulos que se comunican con el módulo Base. Entre las operaciones que realiza se encuentra el armado de paquetes con los payloads requeridos, administración de URLs y enlaces y operaciones con diferentes protocolos soportados.

**Módulos:**

De manera general, llamamos módulos a los paquetes que existen en la base de datos y que se pueden conectar a la Librería Base para agregar alguna funcionalidad a la operación que se está realizando.

* **Exploits:** Un exploit es un recurso de código diseñado para aprovechar alguna vulnerabilidad conocida en un sistema o recurso externo. Típicamente estos exploits permitirán cargar diferentes bloques de código para intentar ejecutarlos en el objetivo aprovechando la vulnerabilidad.
* **Payloads:** El payload o “carga” de un exploit es el código adicional que se quiere ejecutar en el objetivo una vez aprovechada la vulnerabilidad.
* **Encoder:** Los módulos encoder contienen recursos que permiten ofuscar o encubrir el código para que no sea identificado como malicioso por sistemas como antivirus o firewalls.
* **Módulos NOP:** Los módulos NOP permiten agregar cadenas de caracteres en blanco o “0”s en el contenido de un payload. Esto se usa típicamente para ataques de desborde de pila, donde el payload debe tener un tamaño exacto para que la pila apunte exactamente a la posición donde está el código que se desea ejecutar.
* **Módulos auxiliares:** Otras herramientas y recursos que pueden ser requeridos para cargar un exploit o un payload.

**Plugins y herramientas:**

Son otras herramientas o módulos externos a la base de datos de Metasploit, pero que pueden ser requeridos como parte de un proceso o un ataque. Un ejemplo de esto es una implementación de nmap, que típicamente es usada como scanner durante la etapa previa a la identificación de vulnerabilidades.

**Interfaces:**

Estos módulos contienen recursos para que el usuario pueda interactuar con los recursos del framework. Entre estos existen varias interfaces gráficas como Armitage, interfaces web o la consola MSF-Console.

* Console / CLI.
* Interfaces gráficas
* Interfaces web.
* Armitage. Utiliza recursos y cadena de trabajo de un proceso de Metasploit.