

Watch:  **Introducción a Prometheus**

06:23 |

Componentes

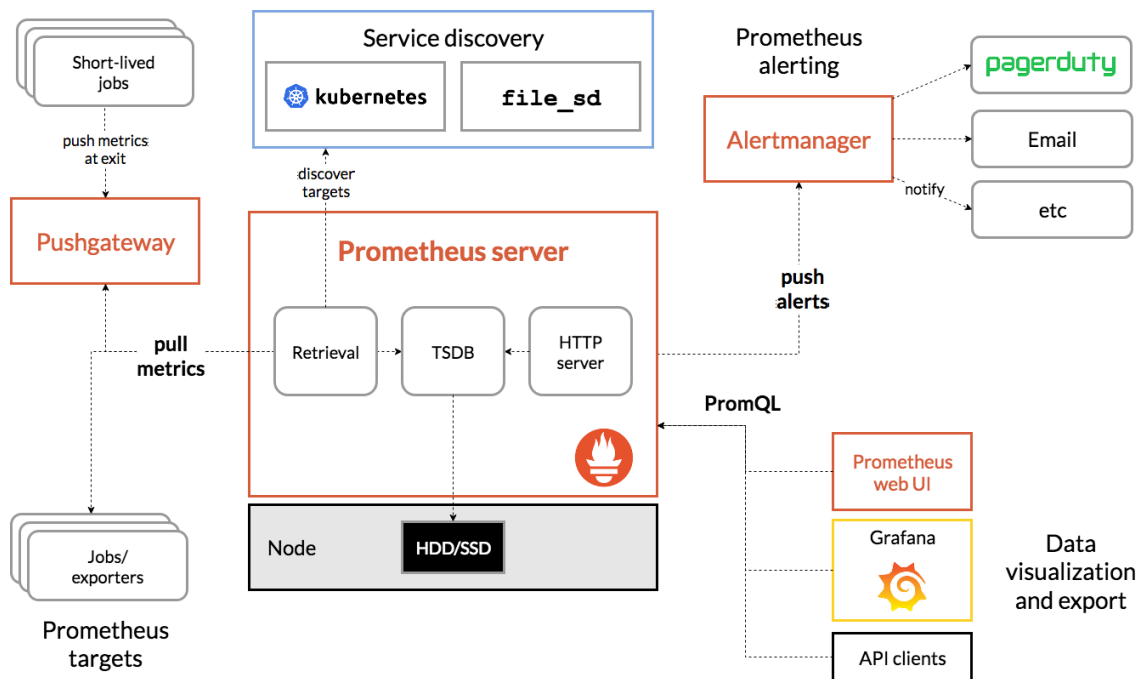
El ecosistema Prometheus consta de múltiples componentes, muchos de los cuales son opcionales:

- El **servidor** Prometheus principal que ingesta y almacena datos de series temporales.
- **Bibliotecas cliente** para instrumentar el código de la aplicación.
- Una **push gateway** para trabajos puntuales.
- **Exportadores** de propósito especial para servicios como HAProxy, StatsD, Graphite, etc.
- Un administrador de **alertas** para manejar alertas: AlertManager.
- Varias herramientas de soporte.

La mayoría de componentes están escritos en lenguaje Go.

Arquitectura

La siguiente imagen (extraída de la documentación oficial) muestra alguna de las piezas fundamentales del ecosistema:



Elementos Operacionales

La filosofía operacional de Prometheus se basa en la **sencillez y eficiencia**. Es por eso que cada Nodo de Prometheus Server es independiente de otro y no existe un *clustering* organizado.

Persistencia

La **persistencia** de los datos en el servidor se hace gracias a una TSDB (*time series database*) built-in dentro del mismo. Cada *sample* de datos

almacenados ocupa 1.3 bytes lo que permite, para organizaciones medianas almacenar información durante meses en un solo servidor.

Desde la versión 1.6 se puede aplicar un modelo de almacenamiento remoto para guardar información más allá de las capacidades de la TSBD.

Escalabilidad

La forma de hacer posible la alta disponibilidad del servicio es simplemente ejecutar dos o más instancias de Prometheus, pero teniendo en cuenta que ninguna de ellas conoce la existencia de las demás.

Service Discovery

Prometheus permite cualquier sistema de *service discovery* (Consul, Zookeeper...) y además provee soporte para proveedores cloud como AWS, Azure y Google.

Cuando es útil

Prometheus funciona bien para grabar cualquier serie temporal numérica. En un mundo de microservicios, su soporte para la recolección y consulta de datos multidimensionales es uno de sus puntos fuertes.

Permite conocer el estado de los diferentes servicios en tiempo real, y en base a ello, crear alertas para notificar a las personas adecuadas en caso de que algo se desvíe de su funcionamiento normal.

Cada servidor Prometheus es independiente, no depende del almacenamiento de red u otros servicios remotos.

Cuando no es útil

Siempre puedes ver qué estadísticas están disponibles sobre su sistema pero si necesitas una precisión del 100%, Prometheus no es una buena opción, ya que los datos recopilados probablemente no serán lo suficientemente detallados y completos.

En tal caso, sería mejor utilizar alguna otra de las técnicas de la observabilidad como el logging o el tracing.