

Monitoreo y trazabilidad

Monitoreo y trazabilidad en la arquitectura integrada con Istio y Kong

Implementación de monitoreo:

En esta arquitectura, el monitoreo se implementa mediante la integración de Istio Service Mesh con herramientas especializadas para captura de métricas, trazabilidad y visualización. Istio inyecta sidecars Envoy en cada pod de los microservicios (SolicitudService y SOAPClientService), lo que permite interceptar y recopilar métricas de red y telemetría sin modificar el código de los servicios.

Los sidecars Envoy envían estas métricas y trazas a sistemas externos configurados para tal fin, proporcionando visibilidad detallada del comportamiento del sistema.

Herramientas utilizadas:

- **Prometheus:** Se utiliza para recopilar y almacenar métricas de rendimiento y estado de los microservicios y del mesh. Prometheus raspa los endpoints de métricas expuestos por Envoy y Kubernetes.
- **Jaeger:** Implementa el trazado distribuido (distributed tracing), permitiendo seguir el recorrido de cada petición a través de los distintos servicios y sidecars, lo que facilita identificar latencias, cuellos de botella y fallos.
- **Kiali:** Proporciona un dashboard visual para observar la topología del service mesh, visualización de tráfico, estado de servicios, políticas aplicadas y problemas detectados. Se integra con Prometheus y Jaeger para mostrar métricas y trazas en un solo lugar.
- **Grafana (opcional):** Para dashboards personalizados y alertas basadas en métricas de Prometheus.

Métricas y trazas capturadas:

- **Métricas de tráfico y latencia:**
 - Tasa de solicitudes (requests per second) por servicio y endpoint.
 - Latencia de respuesta promedio, percentiles y máximos.
 - Tasa y tipo de errores (códigos 4xx, 5xx).
 - Uso de recursos (CPU, memoria) por pod y servicio.
- **Métricas del mesh:**
 - Estado de los circuit breakers y retries configurados.
 - Estado y salud de conexiones mTLS entre servicios.

- **Trazas distribuidas:**

- Path completo de cada petición desde el API Gateway Kong hasta el SolicitudesService y la llamada al SOAPClientService.
- Tiempos de procesamiento en cada microservicio y en la red.
- Identificación de servicios que introducen mayor latencia o generan errores.

Esta implementación permite detectar proactivamente problemas, optimizar la performance y garantizar la seguridad y resiliencia de la plataforma integrada.

Capturas de elementos de monitoreo con kiali

