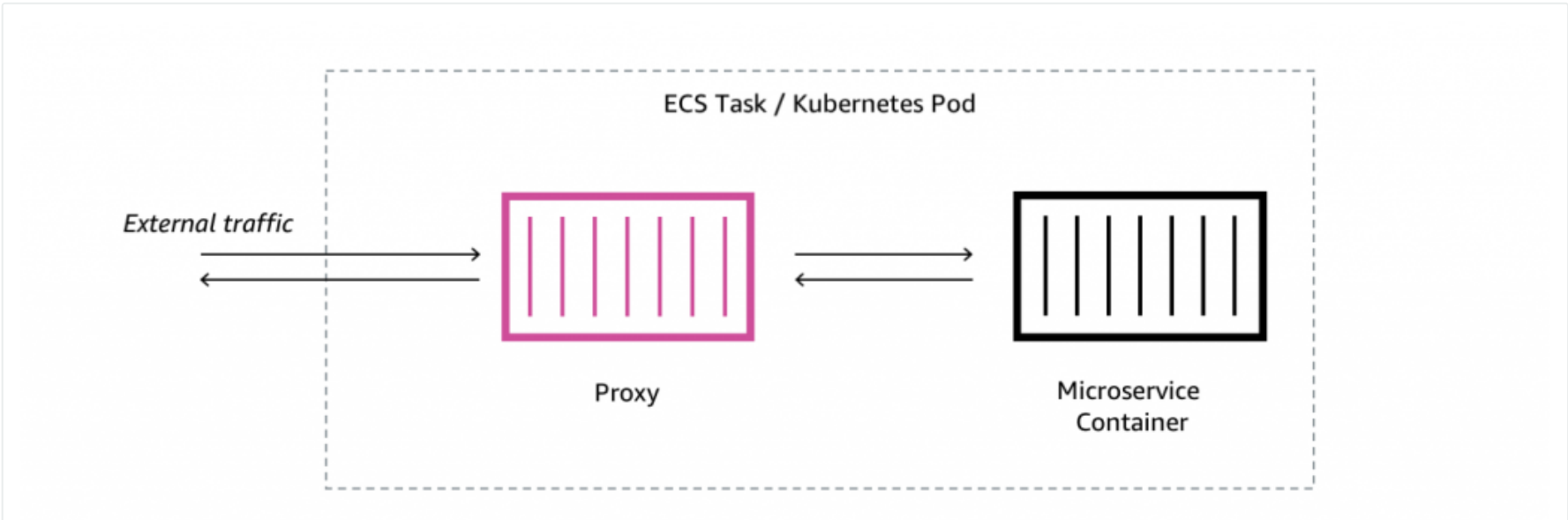
Introduction to App Mesh



App Mesh is a service mesh that provides application-level networking to make it easy for your services to communicate with each other across multiple types of compute infrastructure. App Mesh standardizes how your services communicate, giving you end-to-end visibility and ensuring high-availability for your applications.

Modern applications are typically composed of multiple services. Each service may be built using multiple types of compute infrastructure such as Amazon EC2, Amazon EKS and AWS Fargate. As the number of services grow within an application, it becomes difficult to pinpoint the exact location of errors, re-route traffic after failures, and safely deploy code changes. Previously, this has required you to build monitoring and control logic directly into your code and redeploy your service every time there are changes.

Components

Service mesh – A service mesh is a logical boundary for network traffic between the services that reside within it.

Virtual services – A virtual service is an abstraction of a real service that is provided by a virtual node directly or indirectly by means of a virtual router.

Virtual nodes – A virtual node acts as a logical pointer to a particular task group, such as an ECS service or a Kubernetes deployment. When you create a virtual node, you must specify the service discovery name for your task group.

Envoy proxy – The Envoy proxy configures your microservice task group to use the App Mesh service mesh traffic rules that you set up for your virtual routers and virtual nodes. You add the Envoy container to your task group after you have created your virtual nodes, virtual routers, routes, and virtual services.

Virtual routers – The virtual router handles traffic for one or more virtual services within your mesh.

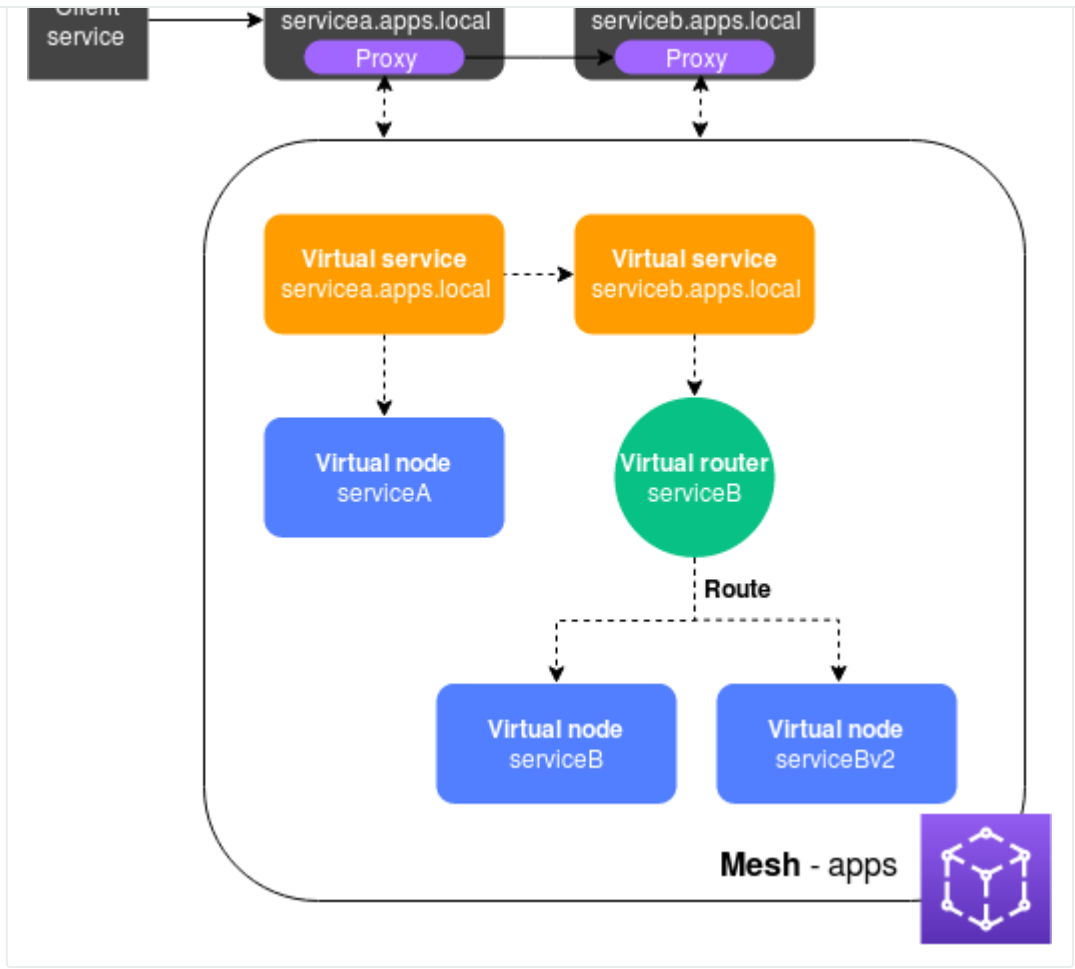
Routes – A route is associated with a virtual router, and it directs traffic that matches a service name prefix to one or more virtual nodes.

Virtual GateWays - A virtual gateway allows resources that are outside of your mesh to communicate to resources that are inside of your mesh. The virtual gateway represents an Envoy proxy running in an Amazon ECS service, in a Kubernetes service, or on an Amazon EC2 instance. Unlike a virtual node, which represents Envoy running with an application, a virtual gateway represents Envoy deployed by itself.

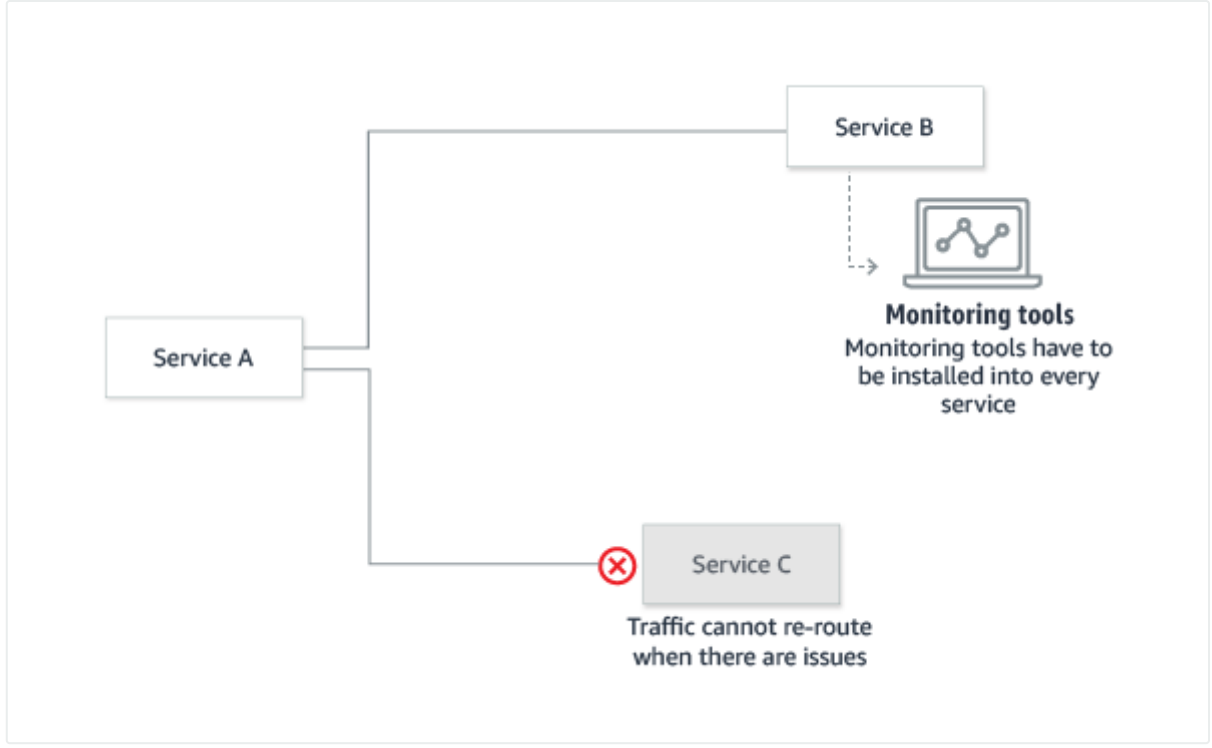
Gateway routes - This is the configuration for handling the incoming requests at the Virtual Gateway. A gateway route is attached to a virtual gateway and routes traffic to an existing virtual service. If a route matches a request, it can distribute traffic to a target virtual service. This topic helps you work with gateway routes in a service mesh.

How it Works

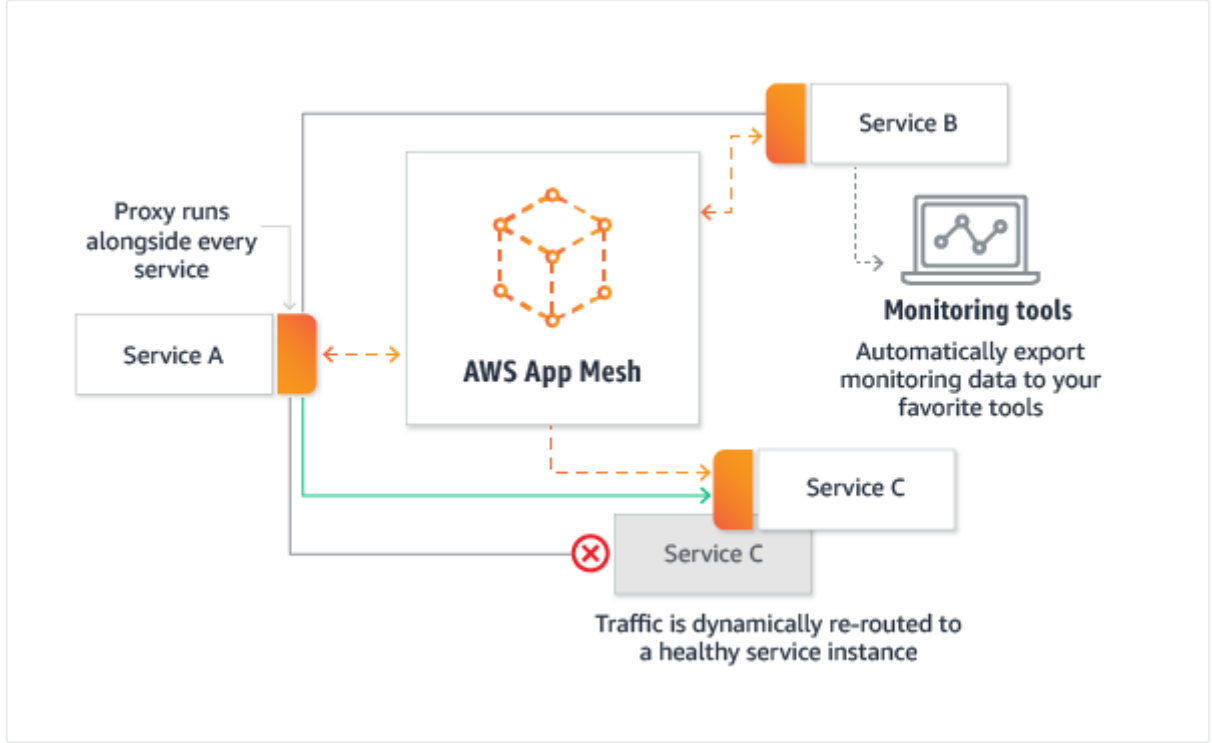
Now let's check out the following image, in this configuration, the services no longer communicate with each other directly. Instead, they communicate with each other through a proxy. The proxy deployed with the servicea.apps.local service reads the App Mesh configuration and sends traffic to serviceb.apps.local or servicebv2.apps.local, based on the configuration.



Before App Mesh - Communications and monitoring are manually configured for every service.



After App Mesh - App Mesh configures communications and monitoring for all services.



Benefits

End-to-end visibility App Mesh captures metrics, logs, and traces from all of your applications. You can combine and export this data to Amazon CloudWatch, AWS X-Ray, and compatible AWS partner and community tools for monitoring and tracing. This lets you quickly identify and isolate issues with any service to optimize your entire application.

Ensure high availability App Mesh gives you controls to configure how traffic flows between your services. You can easily implement custom traffic routing rules to ensure every service is highly available during deployments, after failures, and as your application scales.

Streamline operations App Mesh deploys and configures a proxy that manages all communications traffic to and from your services. This removes the need to configure communication protocols for each service, write custom code, or implement libraries to operate your application.

Enhance any application You can use App Mesh with services running on any compute service such as AWS Fargate, Amazon ECS, Amazon EKS, and Amazon EC2. App Mesh can monitor and control communications for monoliths running on EC2, teams running containerized applications, orchestration systems, or VPCs as a single application without any code changes.

End-to-end Encryption App Mesh gives you the ability to encrypt traffic between services using AWS Certificate Manager (ACM) or customer-provided certificates. This helps you to achieve the security and compliance requirements while building enterprise grade mesh architectures.

Previous Next