Component architecture

Learn more about the Gloo components that you install to manage your environment.

> To review how these components communicate with each other, see the **Networking architecture**. For more information about the management server and agent setup that helps you manage multicluster environments, see the **Relay architecture**.

Component overview

When you install Gloo Mesh Enterprise in your cluster environment, you can set up Gloo management components, optional addons, and Gloo-supported Istio components as described in the following diagram and tables.

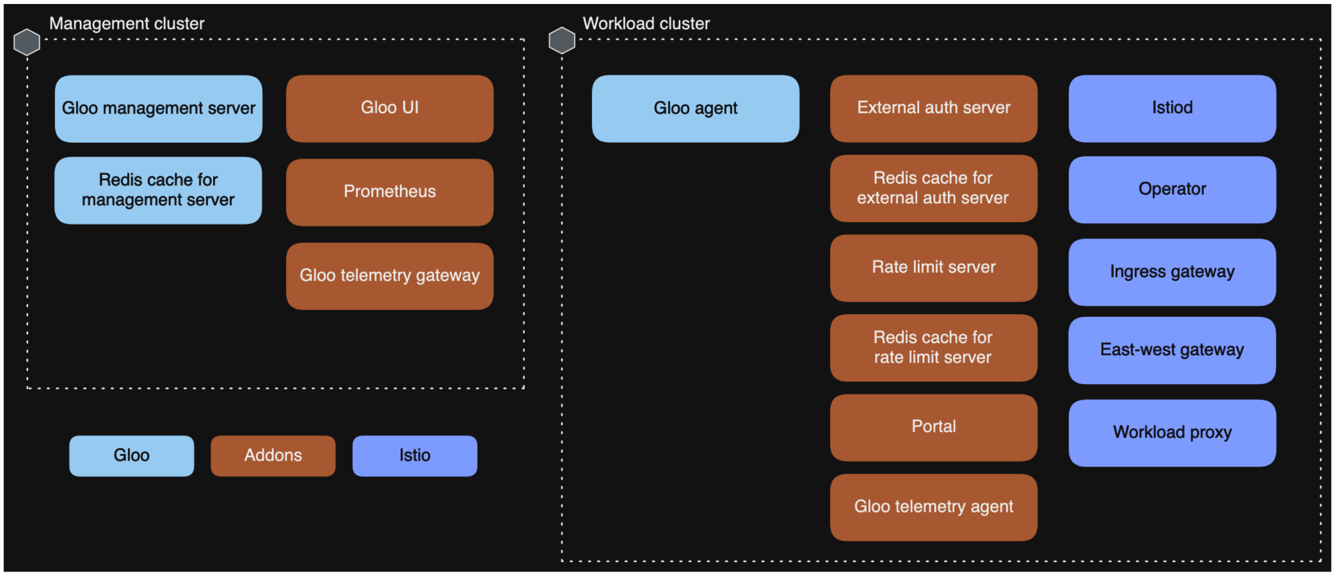

Figure: Gloo management components, addon, and managed Istio components for your cluster environment.

Required Gloo components

By default, Gloo Mesh Enterprise installs the following required components to manage your environment.

Component	Description
Gloo agent	The agents send snapshots of the Gloo resources from each workload cluster to the management server.
Gloo management server	The management server maintains the desired state of your environment based on the configurations that you create. The server translates Gloo custom resources to the appropriate open source custom resources (such as Istio or Envoy). Then, the server pushes config changes to the agents to apply in the workload clusters.
Redis	Redis® * [1] instances are used to store state data for several Gloo components, including the management server, and the state of the custom resources in each registered cluster. You can optionally bring your own Redis instance. If you see state reconciliation errors, you can try restarting Redis.

Optional addons

Install optional addons to extend the Gloo Mesh Enterprise capabilities, such as with rate limiting and external authentication servers.

Component	Description
External auth server	Set up an external authentication and authorization to protect the workloads in your cluster. For example, you can set up basic, passthrough, API key, OAuth, OPA, or LDAP authentication.
Gloo UI	With the UI, you can review the health and configuration of Gloo custom resources, including registered clusters, workspaces, networking, policies, and more. You can even set up external authentication that is synchronized with Kubernetes role-based access control to manage how your users access the UI.
OTel pipeline	You can set up the Gloo OpenTelemetry (OTel) pipeline to collect metrics for your ingress gateway or service mesh.
Portal	With Gloo Portal, you can bundle and secure access to your APIs through a customizable developer portal. The portal supports the OpenAPI specification (OAS), also known as Swagger. Because the APIs must be available externally, Portal works only with Gloo Mesh Gateway.
Prometheus	The default Prometheus deployment scrapes metrics from the Gloo telemetry gateway. You can also bring your own instance.
Rate limit server	Control the rate of requests to destinations within the service mesh.

Redis	Redis instances are used to store state data for several Gloo components. You can optionally bring your own Redis instance. Dashboard: The Gloo UI (dashboard) uses the data in Redis to display resources in the UI. External auth (Gloo Mesh Enterprise, Gloo Mesh Gateway): The external auth server stores its configuration data in a Redis instance that is separate from the one that the management server and dashboard use. Rate limiting (Gloo Mesh Enterprise, Gloo Mesh Gateway): The rate limiting server stores its configuration data in a Redis instance that is separate from the one that the management server and dashboard use.

Gloo-supported Istio components

Component	Description
Istiod	Istiod is the control plane for the Istio service mesh on each workload cluster. For multicluster environments, Gloo federates trust by using a unified root trust policy across clusters.
Ingress gateway	Based on Envoy, the Istio ingress gateway is deployed to manage traffic into and out of the service mesh. Depending on your security requirements, you might set up an ingress gateway per environment, per cluster, or in other ways.
East-west gateway	Based on Envoy, the Istio east-west gateway is deployed in each workload cluster to manage traffic internal to the service mesh, even across clusters. **Note**: When Gloo Mesh Gateway routes incoming requests across clusters through the east-west gateway, the communication from Gloo Mesh Gateway to the east-west gateway is secured with mTLS. However, when your app is deployed without Istio sidecars, the east-west gateway uses plaintext to route the request to your app. To secure communications to your apps with mTLS instead, consider using Gloo Mesh Enterprise alongside Gloo Mesh Gateway to set up an Istio service mesh for your workloads. Additionally, cross-cluster routing through the east-west gateway in Gloo Mesh Gateway is supported only for incoming requests from a client that is external to your cluster environment. You can use Gloo Mesh Enterprise to also route from service-to-service within your cluster environment by using mTLS connections through the east-west gateway.
Workload proxy	Based on Envoy, Istio workload proxies manage network communication between the workload and other microservices. In sidecar mode, each workload

proxy	has its own Istio sidecar proxy for more fine-grained control.

1 * Redis is a registered trademark of Redis Ltd. Any rights therein are reserved to Redis Ltd. Any use by Solo.io, Inc. is for referential purposes only and does not indicate any sponsorship, endorsement or affiliation between Redis and Solo.io. ↩