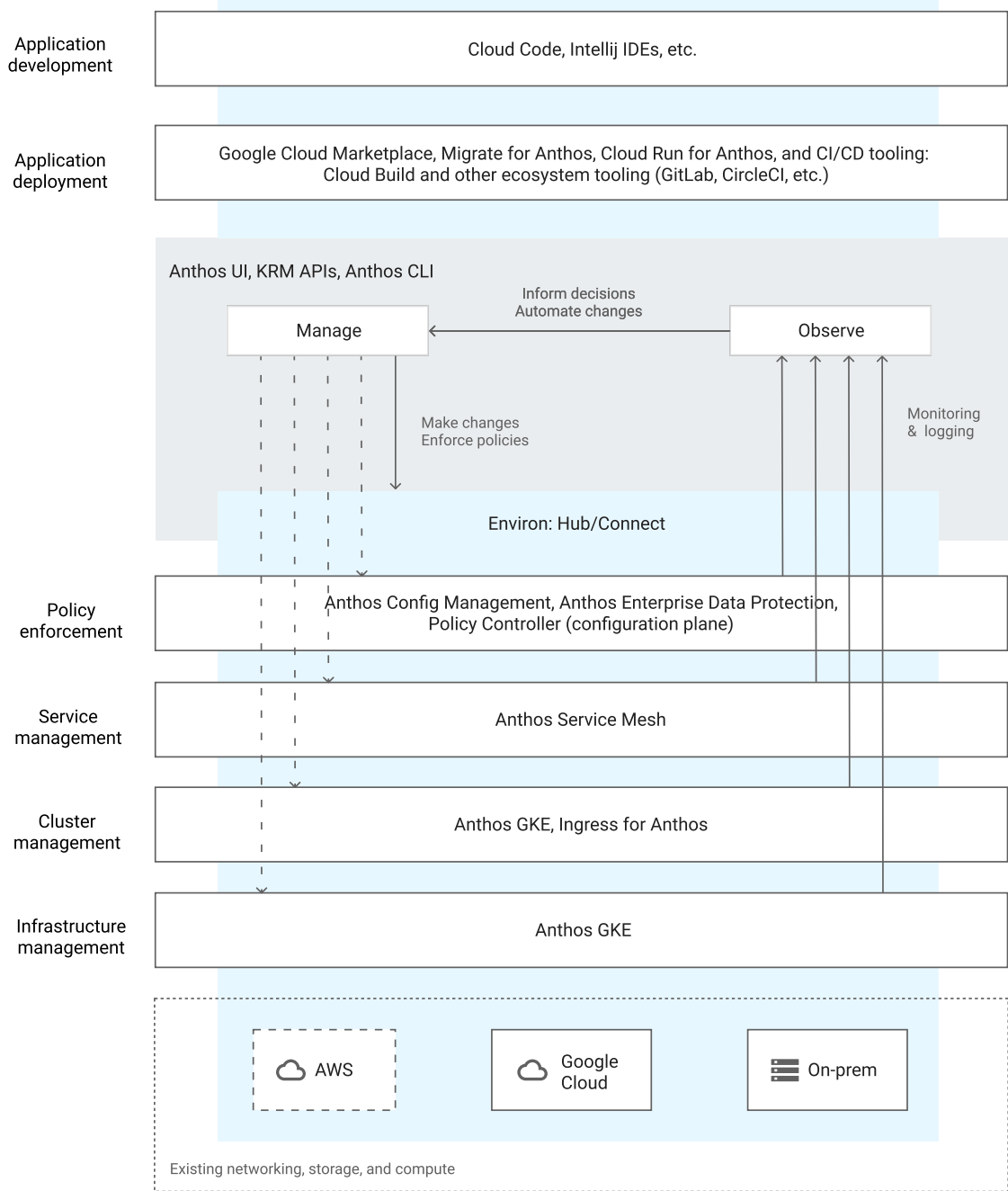


Anthos technical overview

Anthos is a modern application management platform that provides a consistent development and operations experience for cloud and on-premises environments. This page provides an overview of each layer of the Anthos infrastructure and shows how you can leverage its features.

The following diagram shows Anthos components and features and how they provide Anthos's functionality across your environments, from infrastructure management to facilitating application development.



(/anthos/images/anthos-13-components.svg)

The following table shows the components currently available for use on-premises or on Google Cloud. For a full list of the features included in Anthos subscriptions for each deployment option, including product subcomponents, see our [pricing guide](/anthos/pricing) (/anthos/pricing).

Core Anthos components	Cloud	On-premises
Infrastructure, container, and cluster management	Anthos GKE on Google Cloud Ingress for Anthos	Anthos GKE on-prem
Multicluster management	Anthos environ and Connect for Anthos	Anthos environ and Connect for Anthos
Configuration management	Anthos Config Management Policy Controller Config Connector	Anthos Config Management Policy Controller Config Connector
Migration	Migrate for Anthos	
Service management	Anthos Service Mesh (distribution) + Anthos Service Mesh dashboard + Anthos Service Mesh certificate authority	Anthos Service Mesh (distribution)
Serverless	Cloud Run for Anthos	Cloud Run for Anthos
Secure software supply chain	Binary Authorization	
Logging and monitoring	Cloud Logging and Cloud Monitoring for system components	Cloud Logging and Cloud Monitoring for system components
Marketplace	Kubernetes Applications in Google Cloud Marketplace	Kubernetes Applications in Google Cloud Marketplace

Computing environment

The primary computing environment for Anthos relies on [Anthos GKE](/kubernetes-engine) (/kubernetes-engine) on Google Cloud and [Anthos GKE on-prem](/gke-on-prem) (/gke-on-prem) to manage Kubernetes installations in the environments where you intend to deploy your applications. These offerings bundle upstream Kubernetes releases and provide management capabilities for creating, scaling, and upgrading [conformant](https://github.com/cncf/k8s-conformance) (https://github.com/cncf/k8s-conformance) Kubernetes clusters.

Kubernetes has two main parts: the [control plane](https://kubernetes.io/docs/concepts/overview/components/#master-components) (https://kubernetes.io/docs/concepts/overview/components/#master-components) and the [node](#)

components (<https://kubernetes.io/docs/concepts/overview/components/#node-components>). With Anthos on Google Cloud, Google Cloud hosts the control plane, and the Kubernetes API server is the only control-plane component accessible to customers. GKE manages the node components in the customer's project using instances (</compute/docs/instances>) in Compute Engine. With GKE on-prem, all components are hosted in the customer's on-prem virtualization environment.

With Kubernetes installed and running, you have access to a common orchestration layer that manages application deployment, configuration, upgrade, and scaling.

Networking environment

To take advantage of the full functionality provided by Anthos, your Anthos on Google Cloud and Anthos on-prem clusters need IP connectivity among them. Additionally your GKE on-prem environment must be able to reach Google's API endpoints, specifically Cloud Monitoring and Cloud Logging, and Connect for Anthos for registering clusters as part of an Anthos environ, allowing them to be viewed and managed in the Anthos dashboard.

You can connect your on-premises and Google Cloud environments in various ways. The easiest way to get started is by implementing a site-to-site VPN between the environments using Cloud VPN (</vpn/docs>). If you have more stringent latency and throughput requirements, you can choose between Dedicated Interconnect (</interconnect/docs/how-to/choose-type#dedicated>) and Partner Interconnect (</interconnect/docs/how-to/choose-type#partner>). For more information about choosing an interconnect type, see Cloud VPN and other hybrid connectivity solutions (</vpn/docs/concepts/choosing-a-hybrid-solution>).

Kubernetes allows users to provision Layer 4 (<https://kubernetes.io/docs/tasks/access-application-cluster/create-external-load-balancer/>) and Layer 7 (<https://kubernetes.io/docs/concepts/services-networking/ingress/>) load balancers. Anthos on Google Cloud uses Network Load Balancing (</load-balancing/docs/network>) for Layer 4 and HTTP(S) Load Balancing (</load-balancing/docs/https>) for Layer 7. Both are managed services and do not require any additional configuration or provisioning on your part. In contrast, Anthos on-prem uses an on-premises load balancing appliance.

Microservice architecture support

In Kubernetes, services are composed of many Pods (<https://kubernetes.io/docs/concepts/workloads/pods/pod/>), which execute containers. In a microservices

architecture, a single application may consist of numerous services, and each service may have multiple versions deployed concurrently.

With a monolithic application, you have no network-related concerns, because communication happens through function calls that are isolated within the monolith. In a microservice architecture, service-to-service communication occurs over the network.

Networks can be unreliable and insecure, so services must be able to identify and deal with network idiosyncrasies. For example, if Service A calls Service B, and there is a network outage, what should Service A do when it doesn't get a response? Should it retry the call? If so, how often? Or how does Service A know that it is Service B returning the call?

To solve these problems, you can install Anthos Service Mesh on GKE or GKE on-prem. Anthos Service Mesh includes a distribution of [Istio](https://istio.io/docs/concepts/what-is-istio/) (<https://istio.io/docs/concepts/what-is-istio/>), which is an open-source implementation of the service mesh infrastructure layer. Anthos Service Mesh uses sidecar proxies to enhance network security, reliability, and visibility. With Anthos Service Mesh, these functions are abstracted away from the application's primary container and implemented in a common out-of-process proxy delivered as a separate container in the same Pod.

Anthos Service Mesh features include:

- Fine-grained control of traffic with rich routing rules for HTTP(S), gRPC, and TCP traffic.
- Automatic metrics, logs, and traces for all HTTP traffic within a cluster, including cluster ingress and egress.
- Secure service-to-service communication with authentication and authorization based on service accounts.
- Facilitation of tasks like A/B testing and canary rollouts.

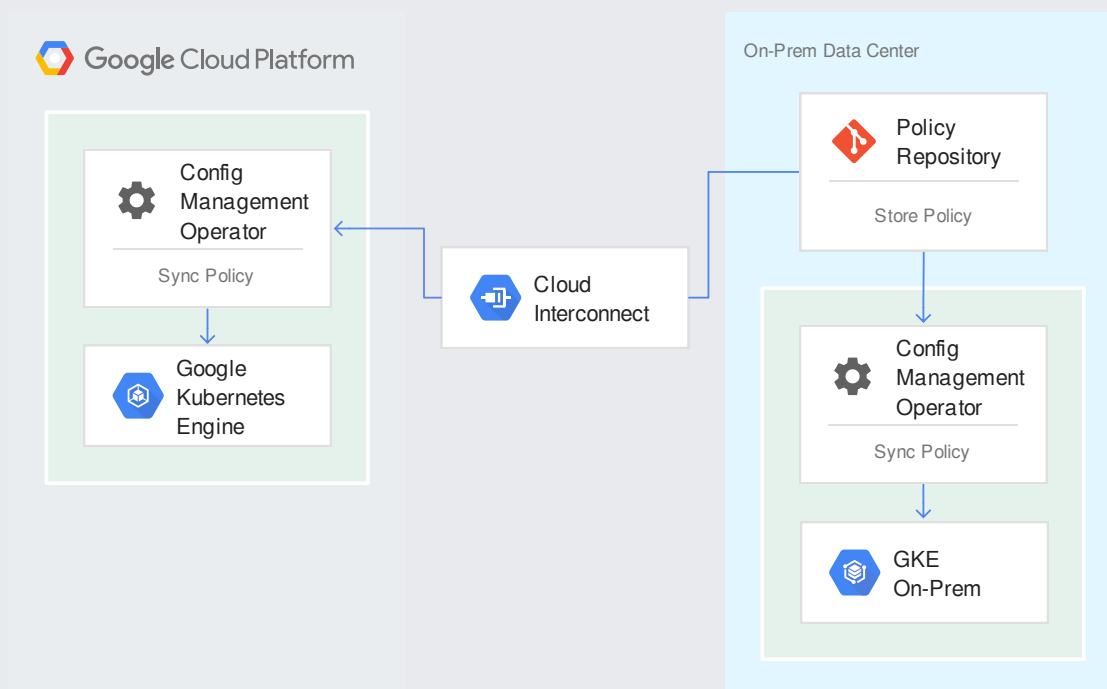
Managed service mesh

For workloads running on Anthos on Google Cloud, Anthos Service Mesh manages your service mesh environment and provides you with many features along with all of Istio's functionality:

- Service metrics and logs for all traffic within your mesh's GKE cluster are automatically ingested to Google Cloud.
- Automatically generated dashboards display in-depth telemetry in the Anthos Service Mesh dashboard, to let you dig deep into your metrics and logs, filtering and slicing your data on a wide variety of attributes.

- Service-to-service relationships at a glance: understand who connects to each service and the services it depends on.
- Secure your inter-service traffic: Anthos Service Mesh certificate authority (Mesh CA) automatically generates and rotates certificates so you can enable mutual TLS authentication (mTLS) easily with Istio policies.
- Quickly see the communication security posture not only of your service, but its relationships to other services.
- Dig deeper into your service metrics and combine them with other Google Cloud metrics using Cloud Monitoring.
- Gain clear and simple insight into the health of your service with service level objectives (SLOs), which allow you to easily define and alert on your own standards of service health.

Centralized config management



(/anthos/images/config-man.svg)

Anthos Config Management architecture (click to enlarge)

Spanning multiple environments adds complexity in terms of resource management and consistency. Anthos provides a unified declarative model for computing, networking, and even service management across clouds and datacenters.

Configuration as data is one common approach to managing this complexity, allowing you to store the desired state of your hybrid environment under version control and apply it directly with repeatable results. Anthos makes this possible with [Anthos Config Management](#) (/anthos-config-management), which integrates with Anthos clusters on-premises or in the cloud. It lets you deploy and monitor configuration changes stored in a central Git repository.

This approach leverages core Kubernetes concepts, such as [Namespaces](#) (<https://kubernetes.io/docs/concepts/overview/working-with-objects/namespaces/>), [labels](#) (<https://kubernetes.io/docs/concepts/overview/working-with-objects/labels/>), and [annotations](#) (<https://kubernetes.io/docs/concepts/overview/working-with-objects/annotations/>) to determine how and where to apply the config changes to all of your Kubernetes clusters, no matter where they reside. The repo provides a versioned, secured, and controlled single source of truth for all of your Kubernetes configurations. Any YAML or JSON that can be applied with [kubectl](#) (<https://kubernetes.io/docs/reference/kubectl/overview/>) commands can be managed with Anthos Config Management and applied to any Kubernetes cluster.

Anthos Config Management also includes the Policy Controller, which enforces custom business logic against every API request to Kubernetes and allows you to use the same logic to scan and audit your clusters. Common security and compliance rules can easily be expressed using the built-in set of rules that deploys with Anthos Config Management, or you can write your own rules using the extensible policy language, based on the open source [Open Policy Agent](#) (<https://www.openpolicyagent.org/>) project.

Anthos Config Management has the following benefits for your GKE clusters:

- Single source of truth, control, and management
 - Enables the use of code reviews, validation, and rollback workflows.
 - Avoids shadows ops, where Kubernetes clusters drift out of sync due to manual changes.
 - Enables the use of CI/CD pipelines for automated testing and rollout.
- One-step deployment across all clusters
 - Anthos Config Management turns a single Git commit into multiple kubectl commands across all clusters.
 - Rollback by simply reverting the change in Git. The reversion is then automatically deployed at scale.
- Rich inheritance model for applying changes
 - Using Namespaces, you can create configuration for all clusters, some clusters, some Namespaces, or even custom resources.

- Using Namespace inheritance, you can create a layered Namespace model that allows for configuration inheritance across the repo folder structure.
- Advanced policy enforcement and auditing with Policy Controller
 - Use the included policy guardrails to enforce security best practices across your entire environment.
 - Continuously audit your environment for configuration that violates business policies.
 - Define and deploy your own custom rules, using an expressive custom policy language to encode your unique business logic.

Serverless

Cloud Run for Anthos provides a developer-focused experience for creating modern applications on the Anthos platform. Cloud Run abstracts away the complexities of the underlying platform, making it easier to generate your own customer value in less time.

Instead of focusing on the platform or authoring lots of YAML, Cloud Run provides clean developer abstractions for defining and deploying services, freeing up the developer to deliver more in less time. Cloud Run manages how the service is run, either in the cloud or on-premises, while optimizing resource utilization, horizontal scaling, and integration with networking and Anthos Service Mesh. Cloud Run can even scale apps to and from zero instances based on demand.

Powered by the [Knative](https://knative.dev) (https://knative.dev) open source project, Cloud Run packages together Google's years of experience running our own serverless platform, and makes it available in your Anthos environment. With Cloud Run, your teams can be faster and focused on meeting customer demand, instead of building out a platform or being locked in to a specific cloud or vendor.

Secure software supply chain

"How do I trust what is running on my production infrastructure?" is one of the top questions we hear from those working in enterprise security and DevOps. Binary Authorization was created to help answer that question. It is a container security feature integrated into Anthos GKE that provides a policy enforcement chokepoint to ensure only signed and authorized images are deployed in your environment.

This capability is especially useful in this era of containerized microservices, as companies often run hundreds to thousands of jobs in production, many of them accessing sensitive and valuable data.

Identity-based deployment control (restricting who can deploy) relies on human operational knowledge and trust that cannot scale to meet the needs of enterprises with automated build and release infrastructure, where deployments may happen hundreds of times a day across dozens of teams.

Binary Authorization formalizes and codifies an organization's deployment requirements by integrating with your chosen CI/CD stages to produce signatures, or "attestations", as an image passes through. Binary Authorization acts as an admission controller by preventing images that do not meet these criteria from being deployed. In addition to signature-based verification, Binary Authorization policy also supports whitelisting images using name patterns. You can specify a repository, a path, or particular set of images that are allowed to deploy.

Binary Authorization is currently available only for Anthos on Google Cloud.

Consolidated logging and monitoring

Access to logs for applications and infrastructure components is critical for running and maintaining production infrastructure. [Cloud Logging \(/logging\)](#) provides a unified place to store and analyze logs. Logs generated by your cluster's internal components are sent automatically to Cloud Logging.

For Anthos on Google Cloud, workloads running inside your clusters have logs automatically enriched with relevant labels like the pod labels, pod name, and cluster name that generated them. Once labeled, logs are easier to explore through [advanced queries \(/logging/docs/view/advanced-filters\)](#).

Also for Anthos on Google Cloud, [Cloud Audit Logs](#)

(<https://cloud.google.com/kubernetes-engine/docs/how-to/audit-logging>) allows you to capture and analyze the interactions that your applications and human users are having with your control components.

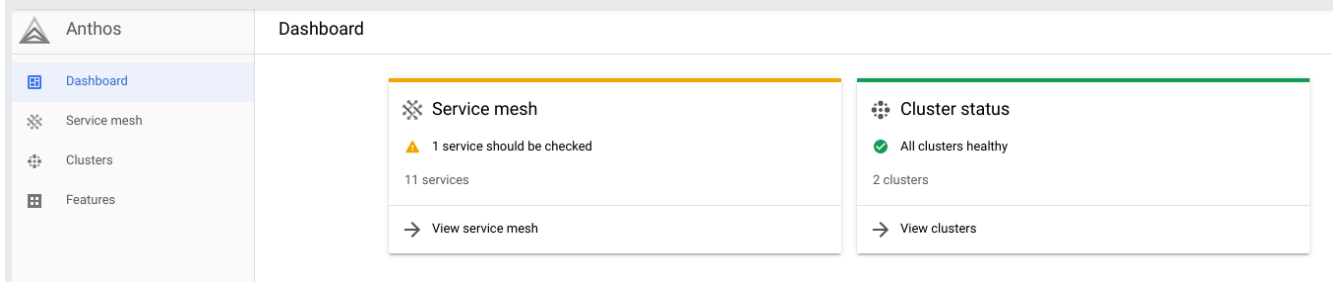
Another key element for operating your Kubernetes environments is the ability to store metrics that provide visibility into your system's behavior and health. [Kubernetes Engine Monitoring \(/monitoring/kubernetes-engine\)](#) provides an automatic and idiomatic integration that stores your application's critical metrics for use in debugging, alerting, and post-incident analysis. For Anthos on Google Cloud, Kubernetes Engine Monitoring is enabled by default.

Anthos includes Cloud Logging and Cloud Monitoring for system components.

Cloud Audit Logs and Kubernetes Engine Monitoring are currently available only for Anthos on Google Cloud but will be for GKE on-prem in future releases.

Unified user interface

The Anthos dashboard in the Google Cloud Console provides you with a secure, unified user interface to view and manage your applications, including an out-of-the-box structured view of all your Anthos resources. The dashboard landing page gives you a runtime status summary for all your services and clusters, and a quick getting started experience for first time users.



Cluster management

As the number of Kubernetes clusters in an organization increases, it can be difficult to understand what is running across each environment. The [Anthos cluster management view](https://console.cloud.google.com/anthos/clusters) (<https://console.cloud.google.com/anthos/clusters>) provides a secure console to view the state of all your registered [clusters](https://kubernetes-engine/docs/concepts/dashboards#kubernetes_clusters) (/kubernetes-engine/docs/concepts/dashboards#kubernetes_clusters) and create new clusters for your project. You add clusters to this view by registering them with Connect for Anthos to your Anthos environ. An environ provides a unified way to view and manage your clusters and their workloads as part of Anthos, including clusters outside Google Cloud.

Service Mesh dashboard

Anthos Service Mesh provides *observability* into the health and performance of your services. Using an adapter, Anthos Service Mesh collects and aggregates data about each service request and response, which means that service developers don't have to instrument their code to collect telemetry data or manually set up dashboards and charts. Anthos Service Mesh automatically uploads metrics and logs to Cloud Monitoring and Cloud Logging for all traffic within your cluster. This detailed telemetry enables operators to observe service behavior, and empowers them to troubleshoot, maintain, and optimize their applications.

← [Service Details Page] Reviews [SEE SERVICE DETAILS](#)

Mesh: Google Prod > Namespace: books-demo > reviews

Filter service

Incidents timeline Showing 9 incidents between 7:00am - 1:30pm 1 hour 6h 12h 1 day 2d 4d 7d 14d 30d [SEE EVENT LIST](#)

OVERVIEW HEALTH METRICS CONNECTED SERVICES DIAGNOSTICS INFRASTRUCTURE CONFIG EXPLORER

Service Status: Unhealthy
There are 5 things that need your attention

4 SLOs

Non-Compliant SLOs
1 !
[See SLO](#)

Trending Non-Compliant SLOs
3 !
[See SLO](#)

Trending Non-Compliant SLOs
1 !
[See SLO](#)

Unauthorized Services
3 !
[Review Certificate](#)

3 Insights

RECOMMENDATION

Your availability SLO target is **too high**. Adjust your SLO target to lower the reliability cost of your service.

→ [See all Metrics](#)

BEST PRACTICE

Your availability SLO target is **too high**. Adjust your SLO target to lower the reliability cost of your service.

→ [See all Metrics](#)

Metrics

<p>Error Rate 0.2% +2% prev 1 hr See Chart</p>	<p>Latency 350 ms +2% prev 1 hr See Chart</p>	<p>Requests 300 /sec +15% prev 1 hr See Chart</p>
--	---	---

Authentication

Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vivamus sagittis lacus vel rutrum faucibus dolor auctor.

Authenticated	8 Services
Unauthenticated	2 Services
Mixed	2 Services

→ [See Inbound/Outbound Services](#)

Topology

Service Details

Project name
DevConsole Test Project GCE

Project ID
google.com:pantheon-testing

Project number
662919498049

→ [See full Details](#)

→ [See Infrastructure](#)

Service Cost

Estimated charges \$3,384

24 instances \$290 (15%)

→ [See Detailed Billing Report](#)

→ [Go to Billing Console](#)

Related Links

→ [See Service Logs](#)

→ [See Config Explorer](#)

→ [Learn more about Managed Services](#)

→ [View more tutorials](#)

On the Service Mesh dashboard, you can:

- Get an overview of all services in your mesh, providing you critical, service-level metrics on three of the four golden signals of monitoring

(https://landing.google.com/sre/sre-book/chapters/monitoring-distributed-systems/#xref_monitoring_golden-signals)

: latency, traffic, and errors.

- Define, review, and set alerts against service level objectives (SLOs), which summarize your service's user-visible performance.
- View metric charts for individual services and deeply analyze them with filtering and breakdowns, including by response code, protocol, destination Pod, traffic source, and more.
- Get detailed information about the endpoints for each service, and see how traffic is flowing between services, and what performance looks like for each communication edge.
- Explore a service topology graph visualization that shows your mesh's services and their relationships.

Currently, the Service Mesh dashboard is supported only for Anthos on Google Cloud clusters, and can't connect to Amazon clusters.

Configuration management

The Anthos dashboard's [Configuration Management view](#)

(https://console.cloud.google.com/anthos/config_management) gives you an at-a-glance overview of the configuration state of all your clusters with Anthos Config Management enabled, and allows you to quickly add the feature to clusters that haven't been set up yet. You can easily track configuration changes and see which branch and commit tag has been applied to each cluster. Flexible filters make it simple to view configuration rollout status by cluster, branch, or tag.

From this view you can also update or install Anthos Config Management on any of your Anthos clusters.

Anthos features

The Anthos dashboard's [Features view](#) (https://console.cloud.google.com/anthos/config_management) lets you view status and manage Anthos features for your registered clusters. Currently you can install Ingress for Anthos on your registered clusters directly from this view. To enable other features, see their [documentation](#) (</anthos/docs/components>).

Third-party application marketplace

The Kubernetes ecosystem is continually expanding and creating a wealth of functionality that can be enabled on top of your existing clusters. For easy installation and management of third-party applications, you can use [Google Cloud Marketplace \(/kubernetes-applications\)](#), which can deploy to your Anthos clusters no matter where they are running. You can also search and filter for Anthos apps and easily find Anthos compatible solutions with the Anthos badge. Google Cloud Marketplace solutions have direct integration with your existing Google Cloud billing and are supported directly by the software vendor.

In the [marketplace solution catalog](#)

(<https://console.cloud.google.com/marketplace/browse?filter=solution-type%3Ak8s>), you'll find:

- Storage solutions
- Databases
- Continuous integration and delivery tools
- Monitoring solutions
- Security and compliance tools

Benefits

Anthos is a platform of distinct services that work in concert to deliver value across the entire enterprise. This section describes how it can be used by the various roles in an organization, and the benefits each group will see.

Anthos for development

For the developer, Anthos provides a state-of-the-art container management platform based on Kubernetes. Developers can use this platform to quickly and easily build and deploy existing container-based applications and microservices-based architectures.

Key benefits include:

- Git-compliant management and CI/CD workflows for configuration as well as code using Anthos Config Management.
- [Code-free instrumentation](https://istio.io/docs/tasks/telemetry/) (https://istio.io/docs/tasks/telemetry/) of code using Anthos Service Mesh and Cloud Monitoring and Cloud Logging to provide uniform observability.
- [Code-free protection of services](https://istio.io/docs/tasks/security/) (https://istio.io/docs/tasks/security/) using mTLS and throttling.

- Support for Google Cloud Marketplace to quickly and easily drop off-the-shelf products into clusters.

Anthos for migration

For migration, Anthos includes Migrate for Anthos, which allows you to orchestrate migrations using Kubernetes in Anthos.

Read about the [benefits of migrating to containers with Migrate for Anthos](/velostrata/docs/anthos-migrate/anthos-migrate-benefits) (/velostrata/docs/anthos-migrate/anthos-migrate-benefits).

Anthos for operations

For Operations, Anthos provides centralized, efficient, and templated deployment and management of clusters, allowing the operations team to quickly provide and manage infrastructure that is compliant with corporate standards.

Key benefits include:

- Centralized configuration management and compliance with configuration-as-code and Anthos Config Management.
- Simplified deployment and rollback with Git check-ins and Anthos Config Management.
- Single pane of glass visibility across all clusters from infrastructure through to application performance and topology.

Anthos for security

For Security, Anthos provides the ability to enforce security standards on clusters, deployed applications, and even the configuration management workflow using a configuration-as-code approach and centralized management.

Key benefits include:

- Centralized, auditable, and securable workflow using Git compliant configuration repos with Anthos Config Management.
- Compliance enforcement of cluster configurations using Namespaces and inherited config with Anthos Config Management.
- Code-free securing of microservices using Anthos Service Mesh and Mesh CA, providing in-cluster mTLS and certificate management.

- Built-in services protection using Anthos Service Mesh authorization and routing.

What's next

Try Anthos

- The [Anthos Sample Deployment on Google Cloud](https://console.cloud.google.com/marketplace/details/click-to-deploy-images/anthos-sample-deployment) (<https://console.cloud.google.com/marketplace/details/click-to-deploy-images/anthos-sample-deployment>) is an Anthos deployment on Google Cloud that lets you explore Anthos features following our [tutorial](https://codelabs.developers.google.com/codelabs/anthos-sample-deployment/) (<https://codelabs.developers.google.com/codelabs/anthos-sample-deployment/>). This is a fully setup deployment with a sample application, and just requires you to create a Google Cloud project.

Learn more

- Learn about [Cloud Interconnect](/interconnect/docs) (/interconnect/docs).
- Learn about [Anthos GKE on-prem](/gke-on-prem) (/gke-on-prem).
- Learn about using [Anthos Service Mesh](/service-mesh/docs/overview) (/service-mesh/docs/overview) to provide a service mesh.
- Learn about [Anthos Config Management](/anthos-config-management) (/anthos-config-management).

Set up Anthos

- Visit our [setup guides](/anthos/docs/setup/overview) (/anthos/docs/setup/overview) to find out how to set up Anthos.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (<https://www.apache.org/licenses/LICENSE-2.0>). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (<https://developers.google.com/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2020-04-03.