Google Cloud

Partner Certification Academy

Professional Cloud Network Engineer

The information in this presentation is classified:

Google confidential & proprietary

⚠ This presentation is shared with you under NDA.

- Do **not** record or take screenshots of this presentation.
- Do **not** share or otherwise distribute the information in this presentation with anyone **inside** or **outside** of your organization.

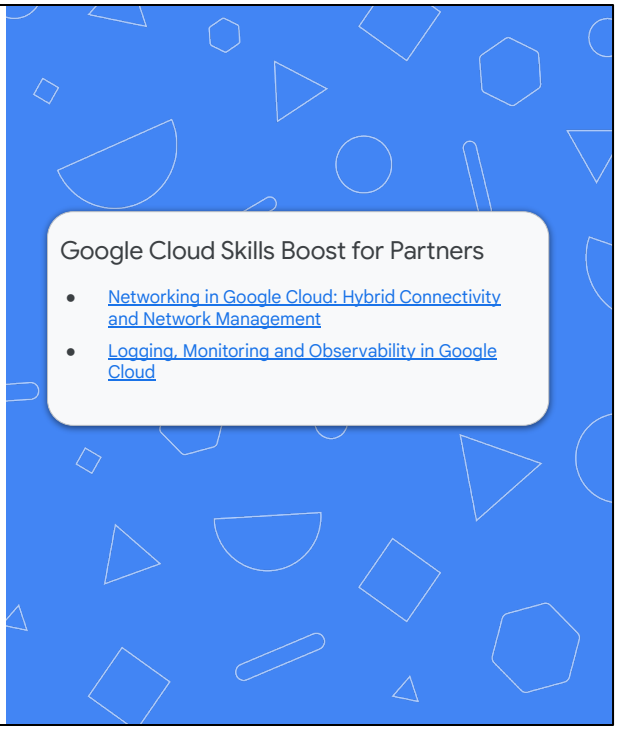
Thank you!

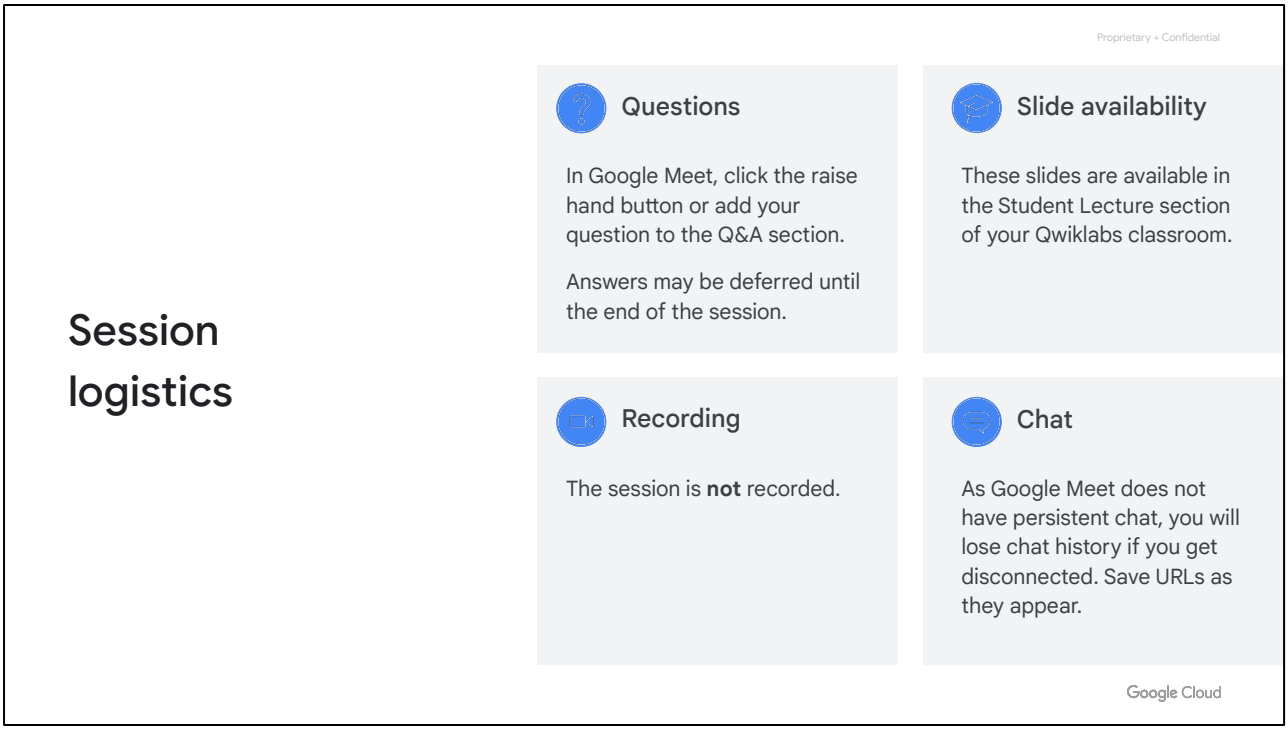
Google Cloud

Source materials

Some of this program's content has been sourced from the following resources:

- Google Cloud certification site
- Google Cloud documentation
- Google Cloud console
- Google Cloud courses and workshops
- Google Cloud white papers
- Google Cloud Blog
- Google Cloud YouTube channel
- Google Cloud partner-exclusive resources





When you have a question, please:
Click the Raise hand button in Google Meet.
Or add your question to the Q&A section of Google Meet.
Please note that answers may be deferred until the end of the session.

These slides are available in the Student Lecture section of your Qwiklabs classroom.

The session is not recorded.

Google Meet does not have persistent chat.
If you get disconnected, you will lose the chat history.
Please copy any important URLs to a local text file as they appear in the chat.

COURSE TITLE

Proprietary + Confidential

Today's agenda

01 Network, billing and pricing

02 Network service tiers

03 Network monitoring and troubleshooting

04 Monitoring critical systems

05 Alerting policies

Google Cloud

AGENDA

Objectives

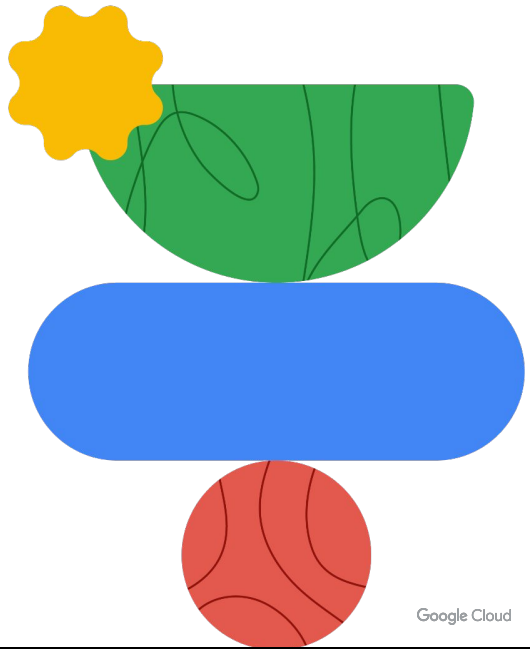
01 Describe network billing and service tiers

02 Understand the monitoring and alerting capabilities available in Google Cloud

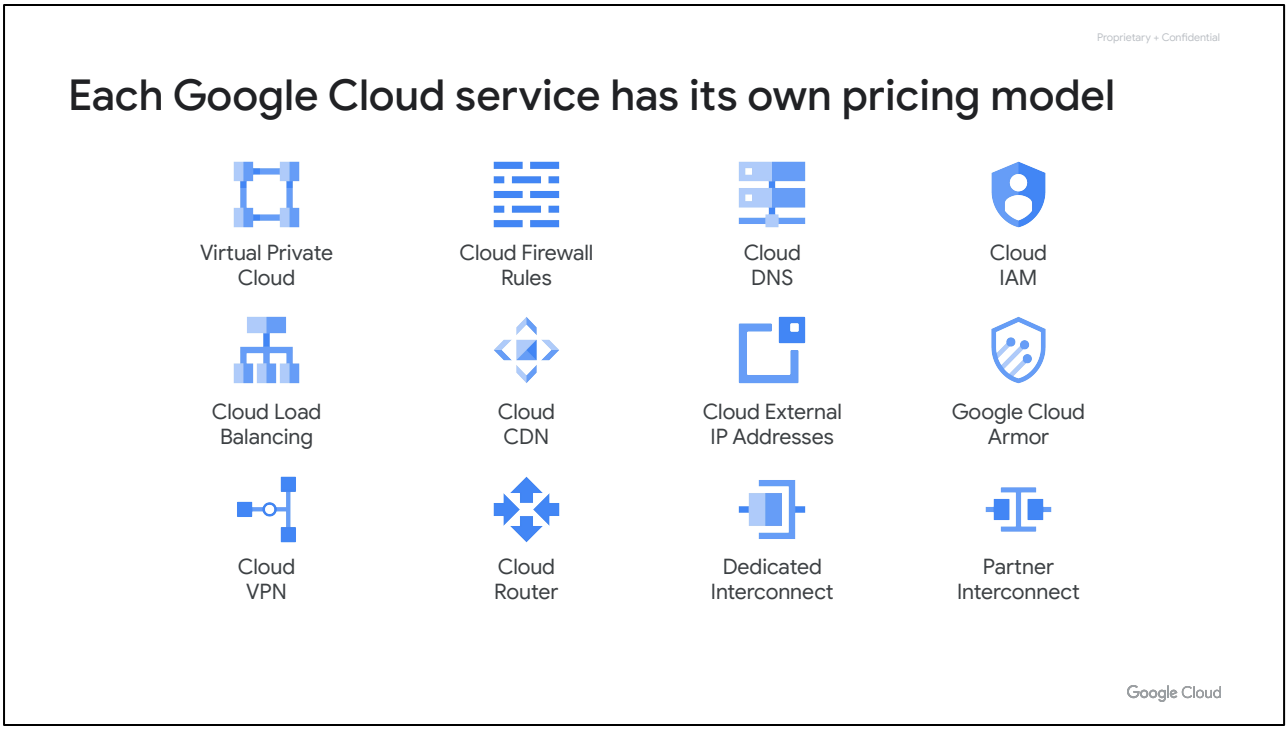
03 Describe some network troubleshooting tools

Objectives

Network, billing and pricing



BREAK SLIDE

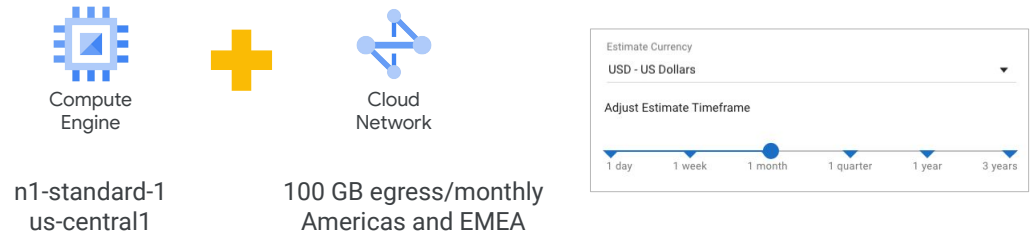


Here are some of the many Google Cloud services that we have covered in this course. Each of these services has its own pricing model and dedicated page within the documentation of each service.

For up-to-date pricing, always refer to the documentation.

There is also a generic Google Cloud price list page that you can navigate to the different services from: https://cloud.google.com/pricing/list

Estimate costs with the Google Cloud Pricing Calculator



https://cloud.google.com/products/calculator/

Because each Google Cloud service has its own pricing model, we recommend using the Google Cloud pricing calculator to estimate the cost of a collection of resources.

The pricing calculator is a web-based tool that you use to specify the expected consumption of certain services and resources, and it then provides you with an estimated cost.

For example, you can specify an n1-standard-1 VM instance in us-central1 along with 100 GB of egress traffic to Americas and EMEA. The pricing calculator then returns the total estimated cost.

You can adjust the currency and time frame to meet your needs and when you are done, you can email the estimate or save it to a specific URL for future reference.

Start using the pricing calculator here https://cloud.google.com/products/calculator/

General network pricing

Traffic type	Price
Ingress	No charge
Egress to the same zone (internal IP address)	No charge
Egress to Google products (YouTube, Maps, Drive)	No charge
Egress to a different Google Cloud service (within same region, exceptions)	No charge
Egress between zones in the same region (per GB)	$0.01
Egress to the same zone (external IP address, per GB)	$0.01
Egress between regions within the US (per GB)	$0.01
Egress between regions, not including traffic between US regions	At internet egress rates

Because pricing can change over time, we will only focus on general network pricing in this module. Specifically, ingress and egress traffic through internal and external IP addresses.
This table is from the Compute Engine documentation, and it lists the price of each traffic type. These rates do not apply for Cloud CDN, CDN Interconnect, Carrier Peering, Direct Peering, and Cloud Interconnect traffic.

First of all, ingress or traffic coming into Google Cloud's network is not charged. The rest of this table lists egress or traffic leaving a Compute Engine instance. Egress traffic that is not charged for is traffic to the same zone, as long as that egress is through the internal IP address of an instance. Also, egress traffic to Google products, like YouTube, Maps, Drive or traffic to a different Google Cloud service within the same region is not charged for either.

However, there is a charge for egress between zones in the same region, egress within a zone if the traffic is through the external IP address of an instance, and egress between regions.

We will cover the internet egress rates shortly because they depend on which Network Service Tier you use.

As for the difference in egress traffic to the same zone, Compute Engine cannot determine the zone of a virtual machine through the external IP address. Therefore, this traffic is treated like egress between zones in the same region.

Also, there are some exceptions, and pricing can always change. For more information, access the Compute Engine pricing in the Google Cloud documentation.

External IP address pricing (us-central1) (Subject to change)

Type	Price/Hour (USD)
Static IP address (assigned but unused)	$0.010
Static and ephemeral IP addresses in use on standard VM instances	$0.004
Static and ephemeral IP addresses in use on preemptible VM instances	$0.002
Static and ephemeral IP addresses attached to forwarding rules	No charge

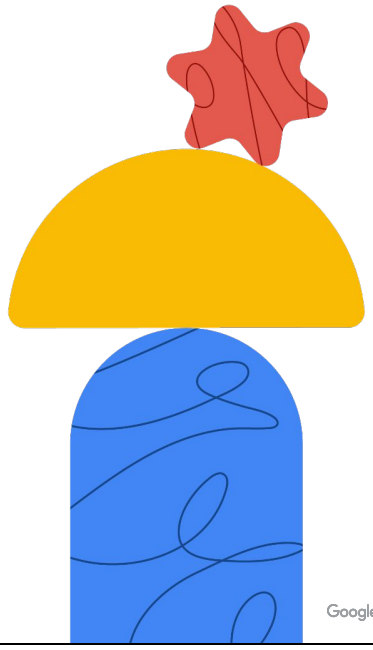
Now, you are charged for static and ephemeral external IP addresses. This table represents the external IP pricing for us-central1 as of this recording.

You can see that if you reserve a static external IP address and do not assign it to a resource, such as a VM instance or a forwarding rule, you are charged at a higher rate than for static and ephemeral external IP addresses that are in use.

Also, external IP addresses on preemptible VMs have a lower charge than for standard VM instances.

Remember, pricing can always change, so please refer to All network pricing in the Google Cloud documentation.

Network Service Tiers



We mentioned earlier that internet egress rates depend on which Network Service Tier you use. So, what are Network Service Tiers?

Use Network Service Tiers to optimize your network for performance or cost

Premium Tier	Standard Tier
High performance routing (Google's network)	Lower price and performance than Premium
Unique to Google Cloud	Comparable to other public cloud offerings
Global SLA	No global SLA
Global load balancing, Cloud CDN	Regional load balancing
Performance is main consideration	Cost is main consideration

Network Service Tiers enable you to optimize your cloud network for performance by choosing Premium Tier or for cost with the new Standard Tier. So what is the difference between these two tiers?

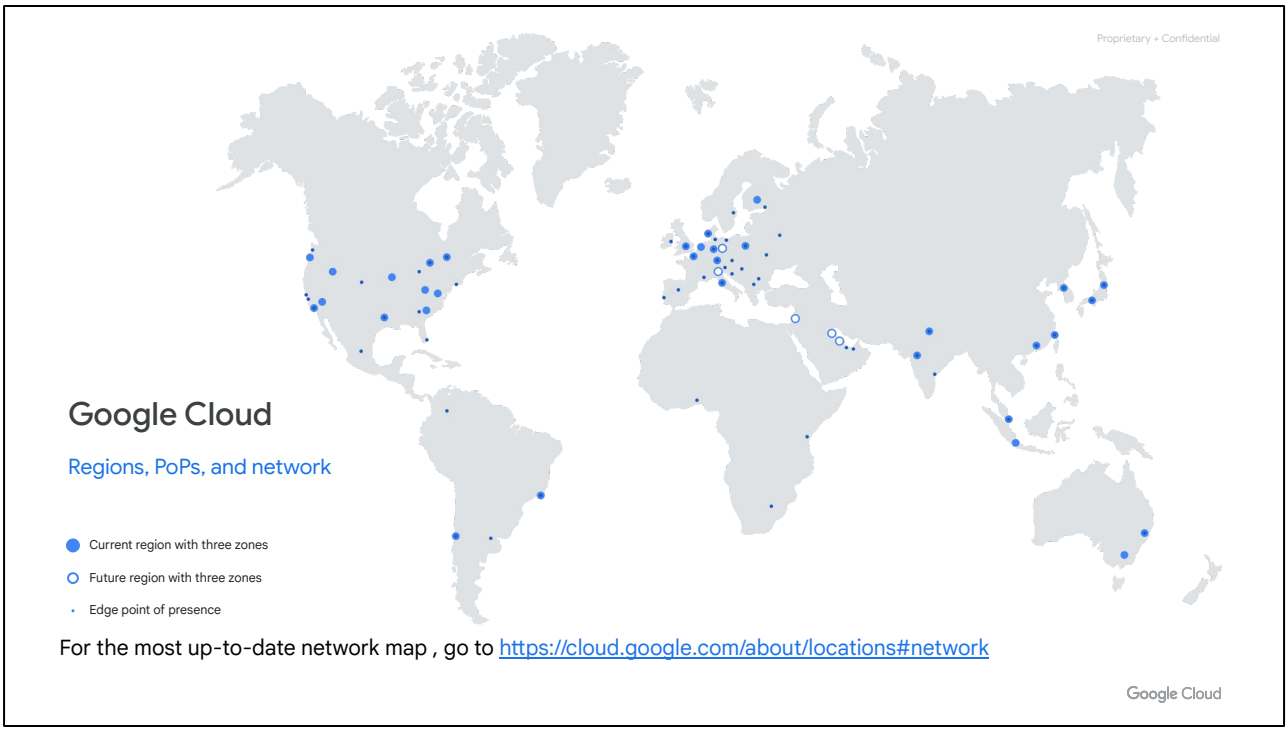
The Premium Tier delivers traffic on Google's global network, providing high-performance routing. If you use Google Cloud today, you already use the powerful Premium Tier. The Standard Tier, alternatively, offers an attractively priced network with performance comparable to that of other major public clouds.

There are other differences between the two tiers: The Premium Tier offers a global SLA and allows for Global Load Balancing and Cloud CDN, as we explored in a previous module. The Standard Tier does not have a global SLA, and Load Balancing is limited to regional.

So why would you choose the Standard Tier? Well, it all comes back to optimizing your cloud network for performance by choosing Premium Tier or for cost with the Standard Tier. In other words, Network Service Tiers allows you to design your cloud network, your way. For performance measurements by Cedexis, please refer to these reports.

Let's explore each Network Service Tier to better understand the network

performance and cost differences.

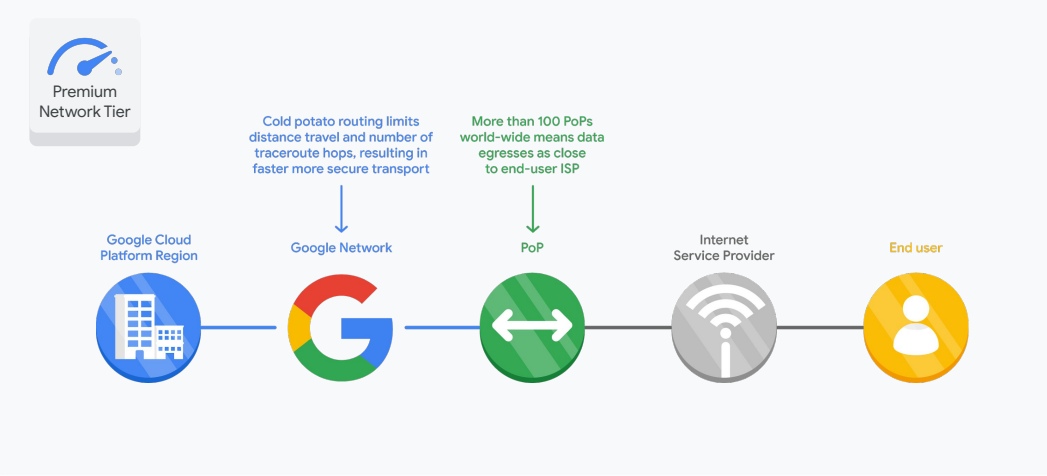


Premium Tier delivers traffic over Google's well-provisioned, low latency, highly reliable global network.

As you can see on this map, this network consists of an extensive global private fiber network with over 100 points of presence across the globe.

Let's explore each Network Service Tier to better understand the network performance and cost differences.

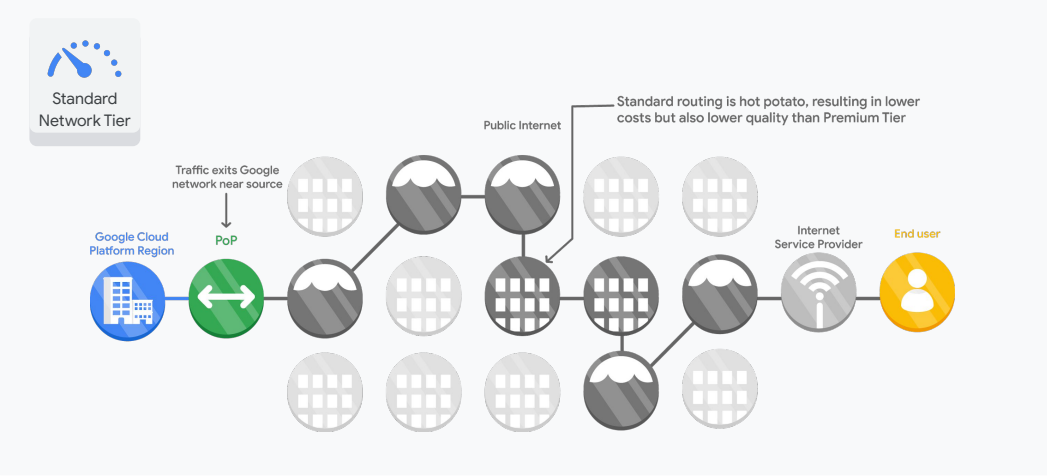
Optimize performance with Premium Tier



In Premium Tier, inbound traffic from your end user to your application in Google Cloud enters Google's private, high performance network at the POP closest to your end user, and Google Cloud delivers this traffic to your application over this network.

Similarly, Google Cloud delivers outbound traffic from your application to end users on Google's network and exits at the POP closest to them, wherever the end users are across the globe. Which means that most of this traffic will reach its destination with a single hop to the end user's ISP, so it enjoys minimum congestion and maximum performance.

Optimize cost with Standard Tier

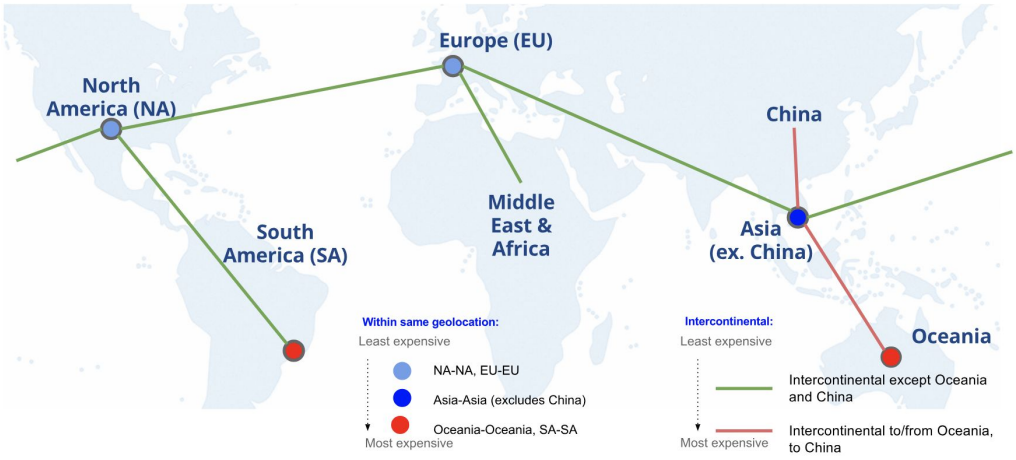


Standard Tier provides network quality that is comparable to other public cloud providers, but lower than Premium Tier. Also, regional network services such as Regional Load Balancing have one VIP per region.

Standard tier is priced lower than Premium because your traffic between Google Cloud and your end user is delivered over ISP networks instead of Google's network.

Now that you understand the differences in performance, let's get into cost.

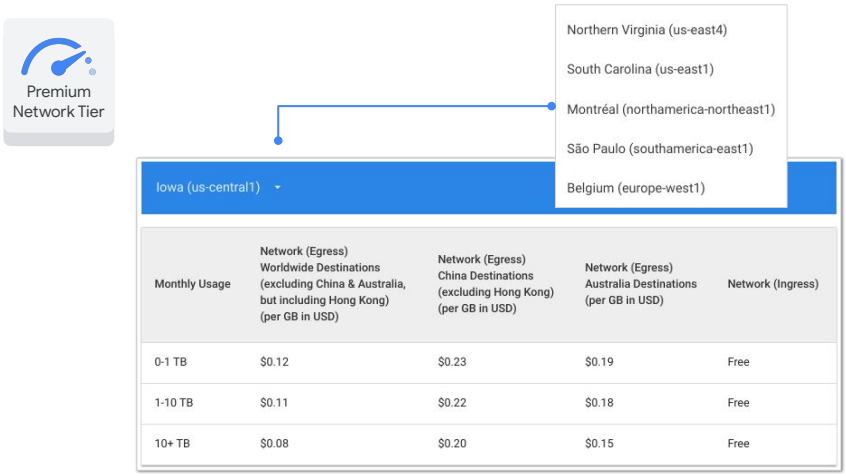
Network Service Tiers pricing



Premium Tier pricing is based on both source and destination of traffic. This is because the cost of network traffic varies with the distance your traffic travels over Google's network. In contrast, Standard Tier traffic is source-based because it does not travel much over Google's network.

This map illustrates that Network Service Tiers categorizes all countries and continents into the listed geolocations. Depending from where to where traffic travels, costs will vary.

Premium Tier pricing

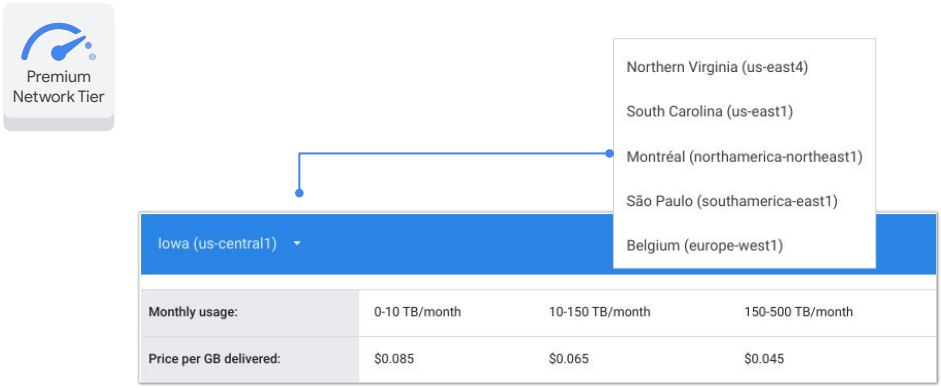


Monthly Usage	Network (Egress) Worldwide Destinations (excluding China & Australia, but including Hong Kong) (per GB in USD)	Network (Egress) China Destinations (excluding Hong Kong) (per GB in USD)	Network (Egress) Australia Destinations (per GB in USD)	Network (Ingress)
0-1 TB	$0.12	$0.23	$0.19	Free
1-10 TB	$0.11	$0.22	$0.18	Free
10+ TB	$0.08	$0.20	$0.15	Free

This Premium Tier pricing table differentiates traffic into 3 monthly usage levels: 0-1 TB, 1-10 TB, and greater than 10 TB. Ingress pricing is free. These prices are applied both during and after the Google Cloud Free Tier period. During the Free Tier period, the prices are charged against the Free Tier credit amount.

For the latest details, and to access the interactive pricing matrix, refer to the pricing documentation: https://cloud.google.com/network-tiers/pricing#premium-pricing

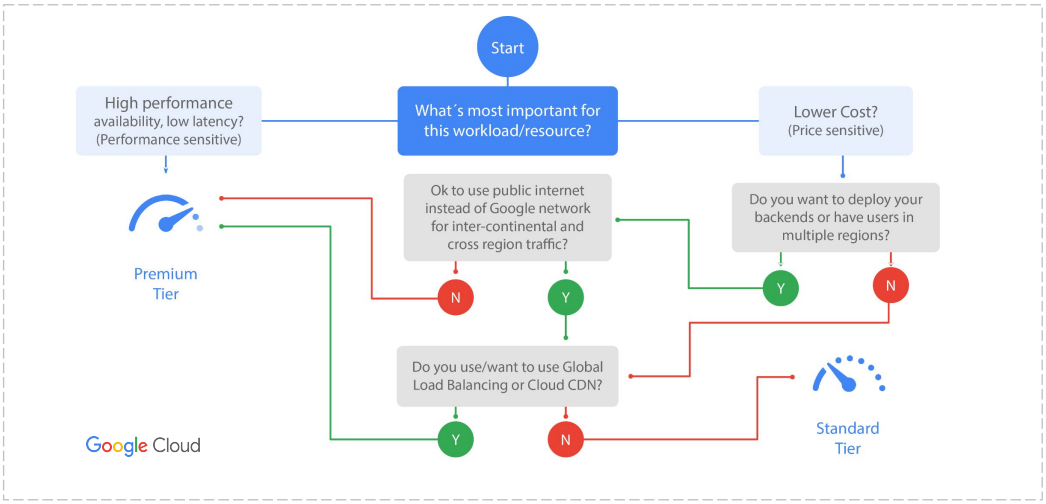
Standard Tier pricing



Similar to the Premium Tier pricing, the Standard Tier pricing model differentiates traffic into 3 monthly usage levels: 0-10 TB, 10-150 TB, and 150-500 TB. If your usage requirement exceeds 500 TB, contact sales for accurate pricing.

For the latest details, and to access the interactive pricing matrix, refer to the pricing documentation. [https://cloud.google.com/network-tiers/pricing#standard-pricing]

Network Service Tiers decision tree



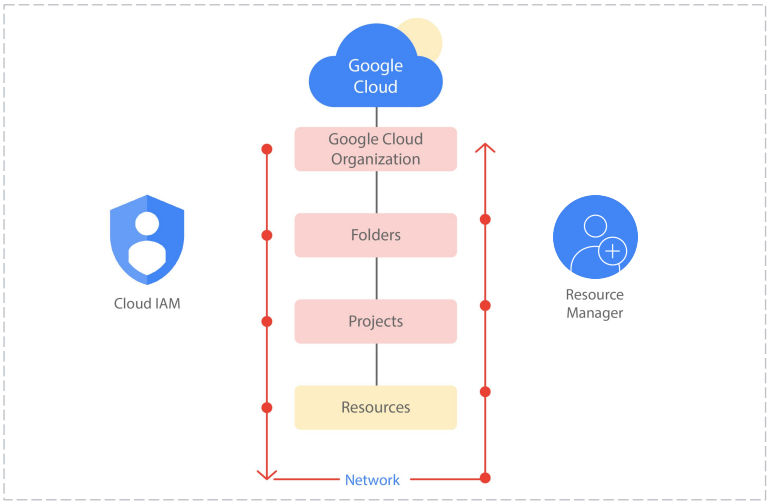
We have gone over both the performance and cost differences between the Network Service Tiers. If you need more guidance on choosing the tier that best meets your needs, use this decision tree.

Ask yourself whether high performance or lower cost is most important for your workload or resource. The Premium Tier is the clear choice for performance. If cost is your main consideration, remember that the Standard Tier has other restrictions in addition to network performance. If you want to deploy your backends or have users in multiple regions, but don't want to use the public internet over Google's network for inter-continental and cross region traffic, you want to choose the Premium Tier.

Also, if you want Global Load Balancing or Cloud CDN, you need to use the Premium Tier.

Otherwise, the Standard Tier is a great choice if you don't need any of those services and are okay using the public internet instead of Google's network.

Billing is accumulated from the bottom up

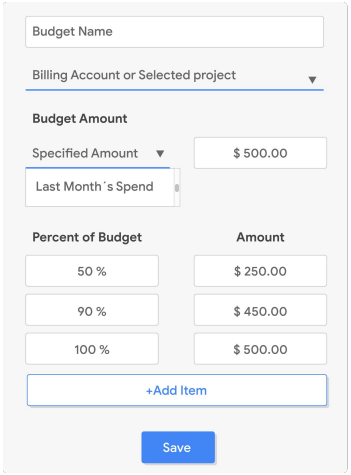


In a previous module, we mentioned that Cloud IAM policies are inherited top-to-bottom, as we can see on the left-hand side. Billing, on the other hand, is accumulated from the bottom up, as we can see on the right-hand side. Because a resource belongs to only one project, a project accumulates the consumption of all its resources.

This is even true for Shared VPC. Billing for resources that participate in a Shared VPC network is attributed to the service project where the resource is located, even though the resource uses the Shared VPC network within the host project.

Now, each project is associated with one billing account, meaning that an organization contains all billing accounts.

Configure budgets and alerts to control costs

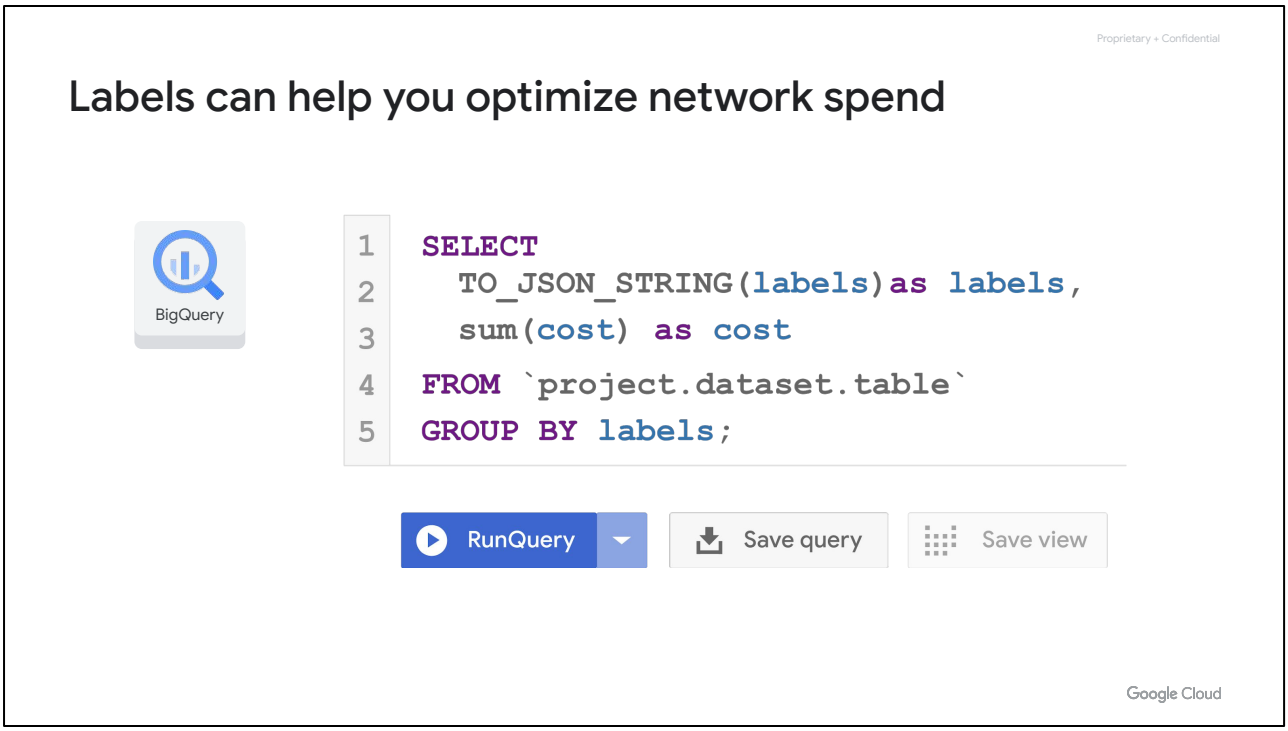


To help you with project planning and controlling costs, you can set a budget in Google Cloud. Setting a budget lets you track how your spend is growing toward that amount.

This screenshot shows the budget creation interface. It allows you to set a budget on either a billing account or a project. You can set the budget at a specific amount or match it to the previous month's spend.

After you determine your budget amount, you can set budget alerts. These alerts send emails to billing admins after spend exceeds a percent of the budget or a specified amount. In our case, it would send an email when spending reaches 50%, 90%, and 100% of the budget amount.

Please note that these alerts are based on estimated expenses, so actual expenses may be greater.



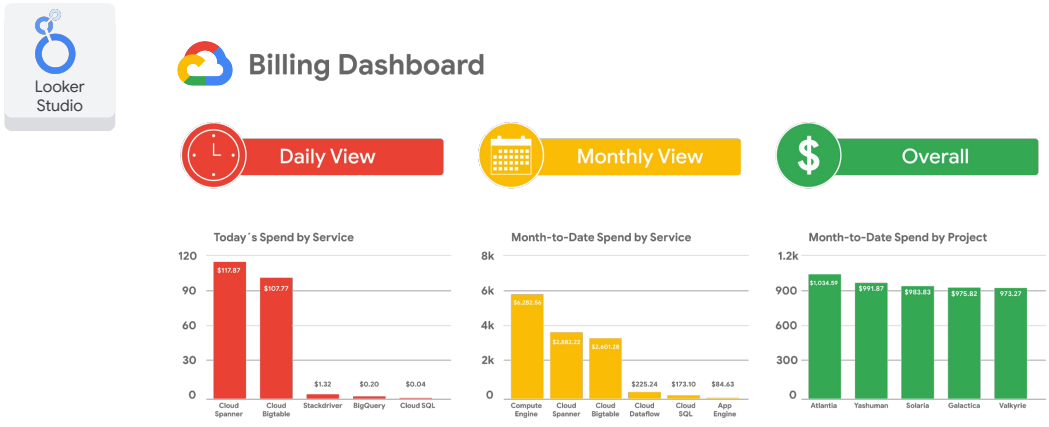
Another way to help optimize your Google Cloud network spend is to use labels. A label is a key-value pair that helps you organize your Google Cloud resources, such as instances. You can attach a label to each resource, which is forwarded to the billing system, so you can break down your billing charges by label.

For example, you could label your backends to better understand the cost of your load balancer. Maybe your backends are sending most of their traffic to a different continent, which could incur higher costs. In that case, you might consider relocating some your backends or using Cloud CDN to cache content closer to your users, which reduces your networking spend.

We recommend labeling all your resources and exporting your billing data to BigQuery to analyze your spend. BigQuery is Google's scalable, fully managed Enterprise Data Warehouse with SQL and fast response times. Creating a query is as simple as shown in this screenshot.

For a demo on how to label your instances and analyze your billing data, please refer here: https://storage.googleapis.com/cloud-training/GoogleCloudnet/student/M6%20-%20Analyze%20billing%20data%20with%20labels.mp4

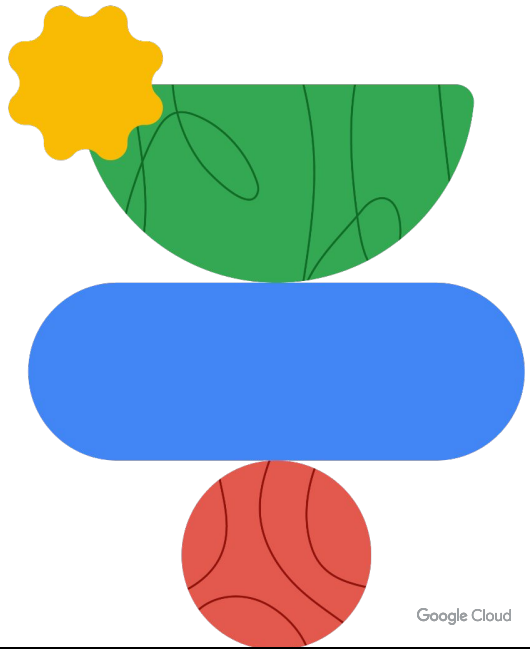
Visualize network spend with Looker Studio



You can even visualize spend over time with Looker Studio. Looker Studio turns your data into informative dashboards and reports that are easy to read, easy to share, and fully customizable.

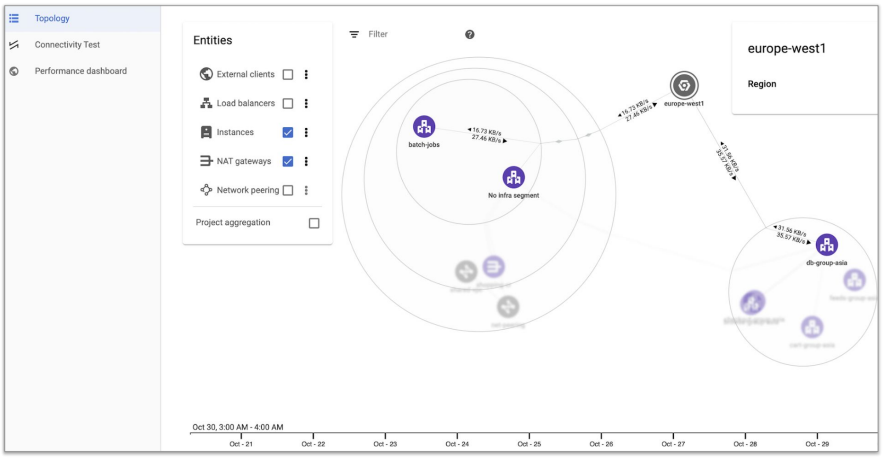
For example, you can slice and dice your billing reports using your labels.

Network monitoring and troubleshooting



BREAK SLIDE

Network Intelligence Center



Seventy five percent of network outages happen due to misconfiguration. More often than not, these misconfigurations are discovered in production.

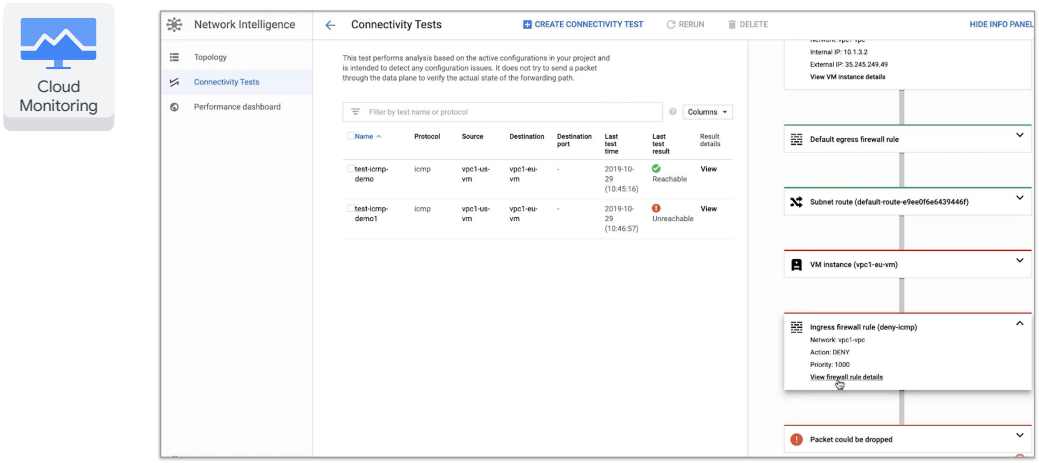
Not knowing the impact of making a configuration change in firewall rules or routing rules makes network monitoring reactive rather than proactive, introducing risk and extending mean time to resolution.

Network Intelligence Center enables teams to prevent networking outages and performance issues before they happen.

Centralized monitoring cuts down troubleshooting time and effort, increases network security, and improves the overall user experience.

Network Intelligence Center modules offer network topology visualization, network connectivity tests, a performance dashboard and firewall insights.

Diagnose issues using Connectivity Test



Google Cloud

When a virtual machine is unreachable, you need to diagnose the connectivity issue quickly to prevent any issues. For example, you may have an issue between source and destination endpoints in your VPC network.

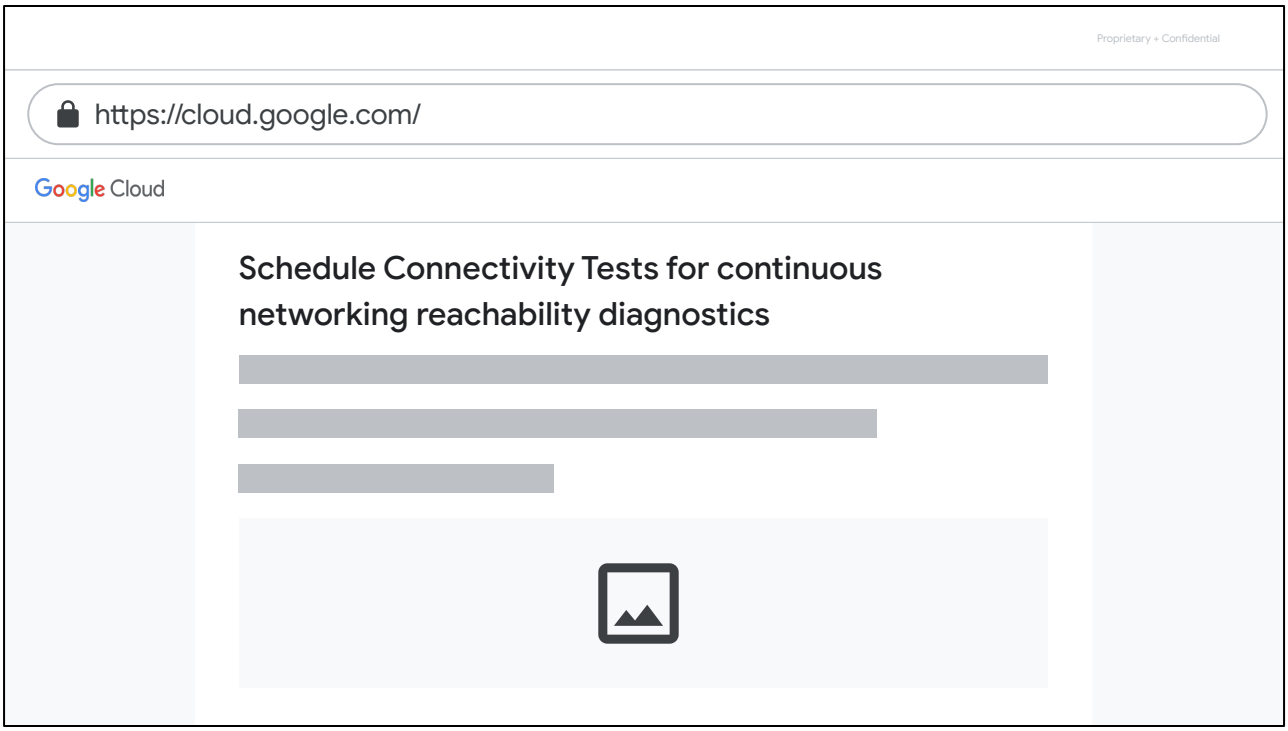
Using the Network Intelligence Center Connectivity Test, you can self-diagnose connectivity issues within Google Cloud or Google Cloud to an external IP address which could be on-prem or another cloud, helping to isolate whether the issue is in Google Cloud or not.

You can create, save and run tests to help verify the impact of configuration changes and ensure that network intent captured by these tests is not violated, proactively preventing network outages.

These tests also help assure network security and compliance. Connectivity Test has been used internally by the Google Cloud support team to resolve customer issues.

Connectivity tests and reachability diagnostics

- Schedule Cloud function to run regular connectivity tests
- Any connectivity failures can be logged to Cloud Logging
- Log-based alert can be fired to notify administrators



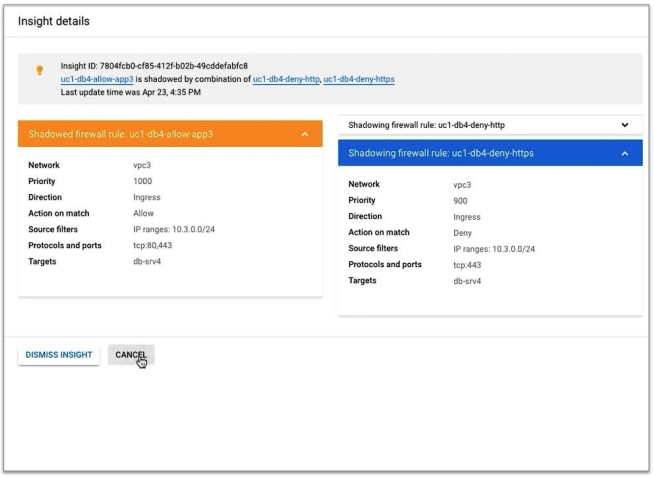
Read the blog titield ‘Schedule Connectivity Tests for continuous networking reachability diagnostics’ for more information.

Performance dashboard

How can you diagnose if the application or the underlying network is the root cause or your issues?

Network Intelligence Center's Performance Dashboard can show you real-time performance metrics (latency and packet loss) between the zones where you have VMs, enabling you to quickly troubleshoot where the packet loss is happening, and indeed, if it's a networking issue at all.

Firewall insights



Firewall configuration can be daunting. How can you verify that firewall rules are being used in the intended way?

Firewall Insights enables you to better understand and safely optimize your firewall configurations.

Firewall Insights provides reports that contain information about firewall usage and the impact of various firewall rules on your Virtual Private Cloud (VPC) network.

Firewall insights helps you better understand and safely optimize your firewall rules

Firewall Insights, a component product of Network Intelligence Center, produces metrics and insights that let you make better decisions about your firewall rules. It provides data about how your firewall rules are being used, exposes misconfigurations, and identifies rules that could be made more strict.

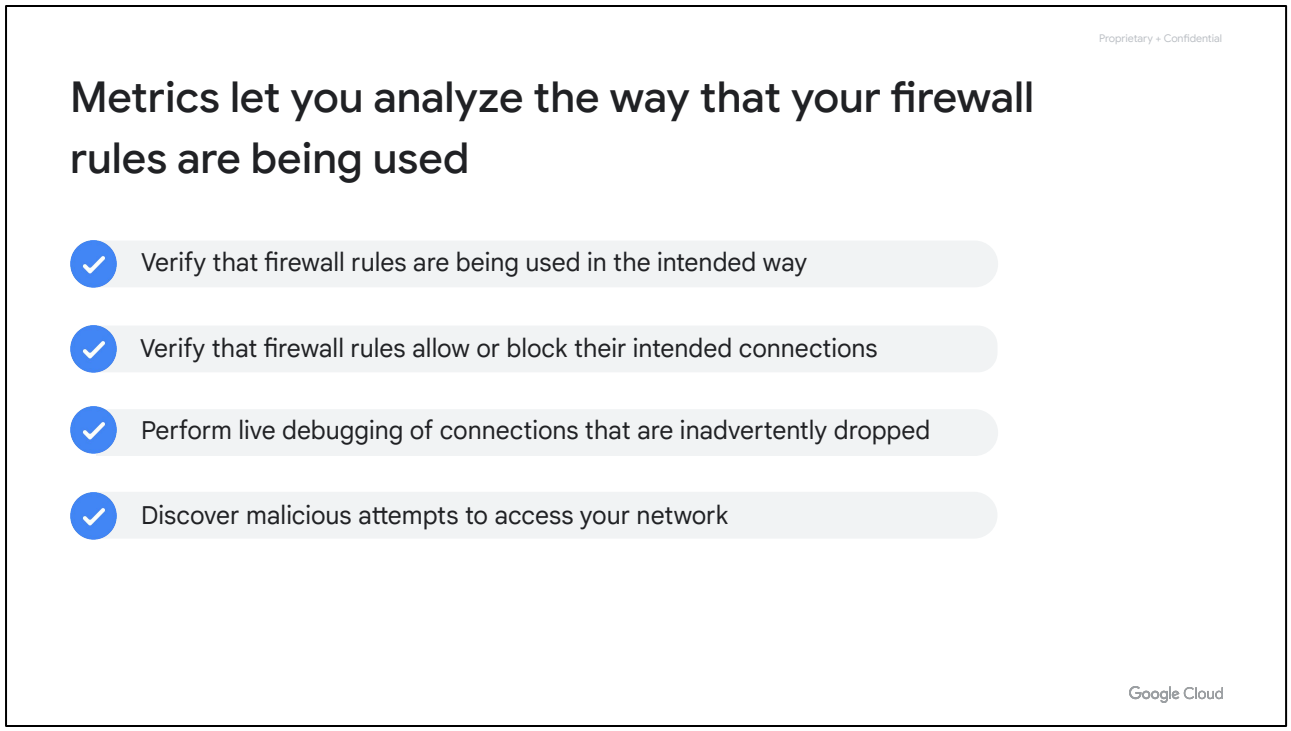
Firewall Insights uses Cloud Monitoring metrics and Recommender insights.

Cloud Monitoring collects measurements to help you understand how your applications and system services are performing. A collection of these measurements is generically called a metric. The applications and system services being monitored are called monitored resources. Measurements might include the latency of requests to a service, the amount of disk space available on a machine, the number of tables in your SQL database, the number of widgets sold, and so forth. Resources might include virtual machines, database instances, disks, and so forth.

Recommender is a service that provides recommendations and insights for using resources on Google Cloud. These recommendations and insights are per-product or per-service, and are generated based on heuristic methods, machine learning, and current resource usage. You can use insights independently from recommendations. Each insight has a specific insight type. Insight types are specific to a single Google Cloud product and resource type. A single product can have multiple insight types, where each provides a different type of insight for a different resource.

Using Cloud Monitoring for metrics:

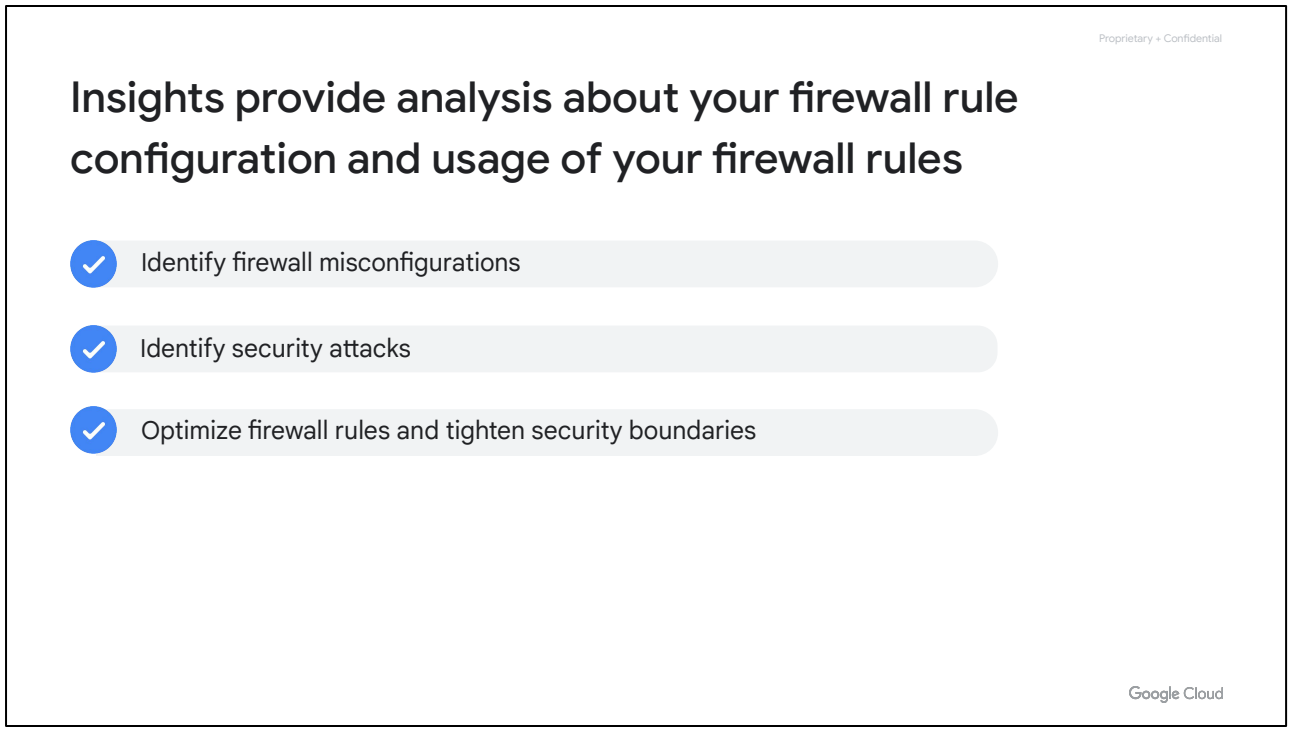
https://cloud.google.com/monitoring/api/v3/metrics
Using Recommender for insights:
https://cloud.google.com/recommender/docs/insights/using-insights



Firewall Insights metrics let you analyze the way that your firewall rules are being used. Firewall Insights metrics are available through Cloud Monitoring and the Google Cloud Console. Metrics are derived through Firewall Rules Logging.

With Firewall Insights metrics, you can perform the following tasks:

- Verify that firewall rules are being used in the intended way.
- Over specified time periods, verify that firewall rules allow or block their intended connections.
- Perform live debugging of connections that are inadvertently dropped because of firewall rules.
- Discover malicious attempts to access your network, in part by getting alerts about significant changes in the hit counts of firewall rules.



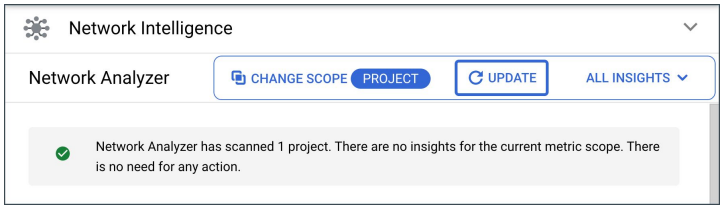
Insights provide analysis about your firewall rule configuration and usage of your firewall rules. They use the `google.compute.firewall.Insight` insight type.

With insights, you can perform the following tasks:

- Identify firewall misconfigurations.
- Identify security attacks.
- Optimize firewall rules and tighten security boundaries by identifying overly permissive allow rules and reviewing predictions about their future usage. Please note that at the time of writing, these capabilities are in preview.

Network Analyzer

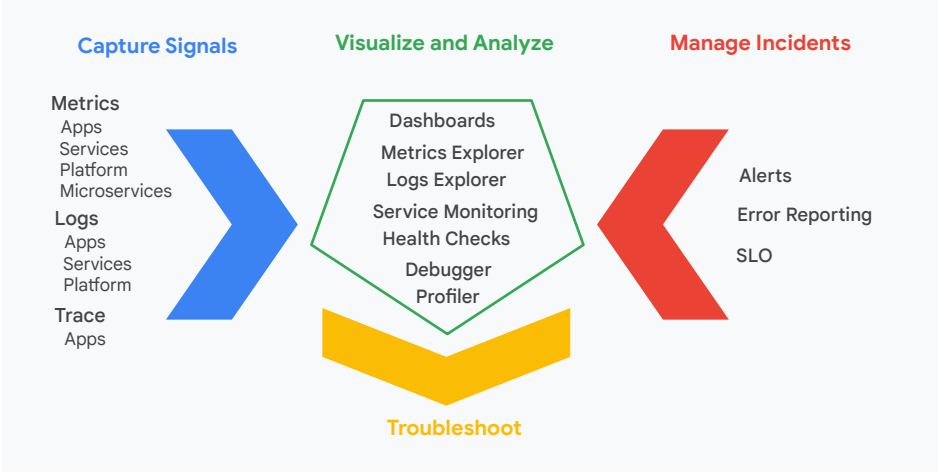
- Automatically monitors your VPC network configurations and detects misconfigurations and suboptimal configurations.
- Provides insights on network topology, firewall rules, routes, configuration dependencies, and connectivity to services and applications.
- Identifies network failures, provides root cause information, and suggests possible resolutions.



Network Analyzer automatically monitors your VPC network configurations and detects misconfigurations and suboptimal configurations. It provides insights on network topology, firewall rules, routes, configuration dependencies, and connectivity to services and applications. It identifies network failures, provides root cause information, and suggests possible resolutions.

Network Analyzer runs continuously and triggers relevant analyses based on near real-time configuration updates in your network. If a network failure is detected, it tries to correlate the failure with recent configuration changes to identify root causes. Wherever possible, it provides recommendations to suggest details on how to fix the issues.

Google Cloud observability



Next, let's explore Google Cloud monitoring tools. If you've ever worked with on-premises environments, you know that you, or someone in your organization, can actually lay hands on any of your servers. If an application becomes unresponsive, someone can walk in and physically check as to why.

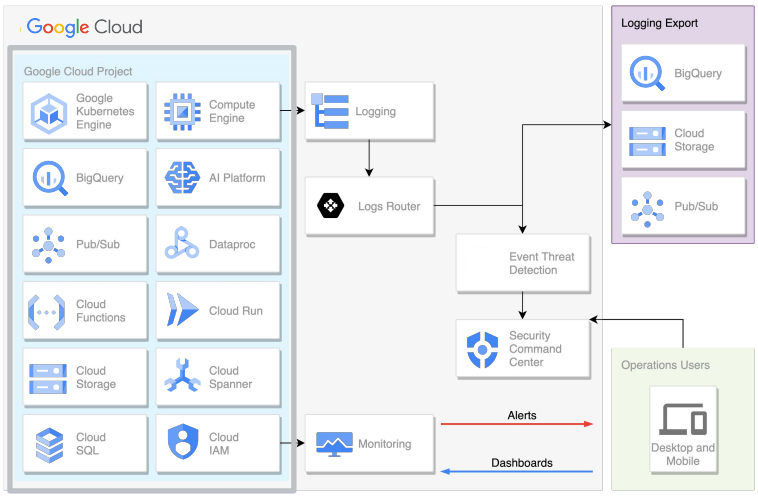
In the cloud though, the servers aren't yours, they're Google's, and you aren't going to be able to inspect them physically. So the question becomes, how do you know what's happening with your server, or database, or application? The answer is the tools discussed in this course.

It all starts with signals. Metric, logging, and trace data capturing is integrated into Google products from the hardware layer up. From those products, the signal data flows into the Google Cloud operation's tools where it can be visualized in Dashboards and through the Metrics Explorer. Automated and custom logs can be dissected and analyzed in the Log Viewer. Services can be monitored for compliance with Service Level Objectives (SLOs), and error budgets can be tracked. Health Checks can be used to check uptime and latency for external-facing sites and services. And running applications can be debugged and profiled.

When Incidents occur, signal data can generate automated Alerts to code or, through various information channels, to key personnel. Error Reporting can help operations and developer teams spot, count, and analyze crashes in cloud-based services. The visualization and analysis tools can then help troubleshoot what's happening in Google Cloud.

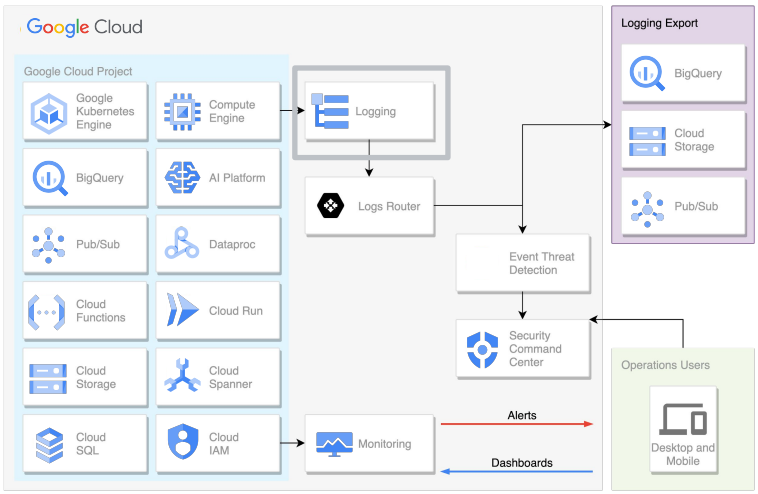
Ultimately, you won't miss that easy server access, because Google is going to allow you more precise insights into your Cloud install than you ever had on-premises.

Application and infrastructure observability



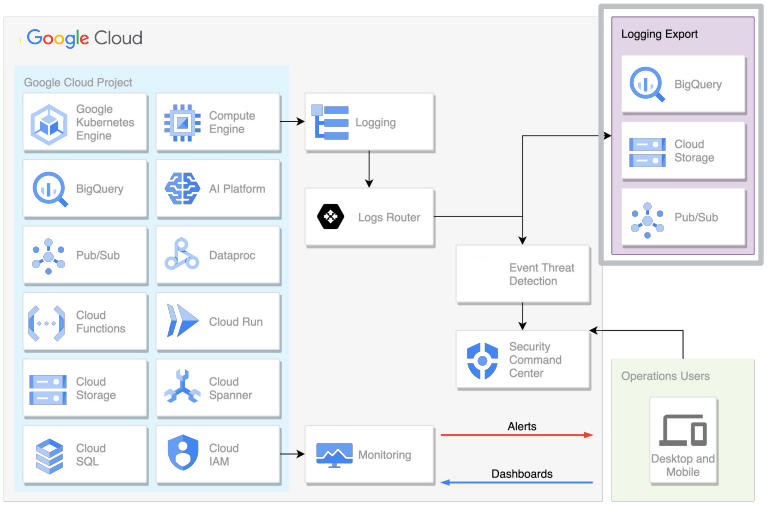
Google Cloud has many products, from Kubernetes, to BigQuery, to Spanner, and they all stream metrics and logs into Google's Cloud Logging and Cloud Monitoring components.

Application and infrastructure observability



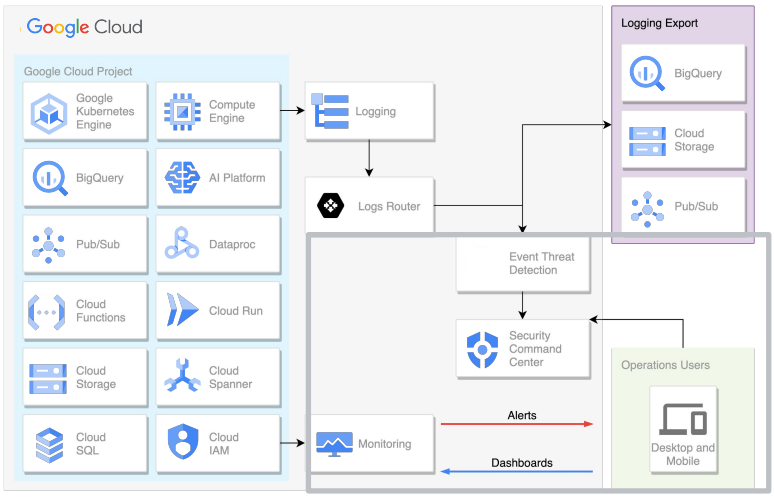
Google Cloud has many products, from Kubernetes, to BigQuery, to Spanner, and they all stream metrics and logs into Google's Cloud Logging and Cloud Monitoring components.

Application and infrastructure observability



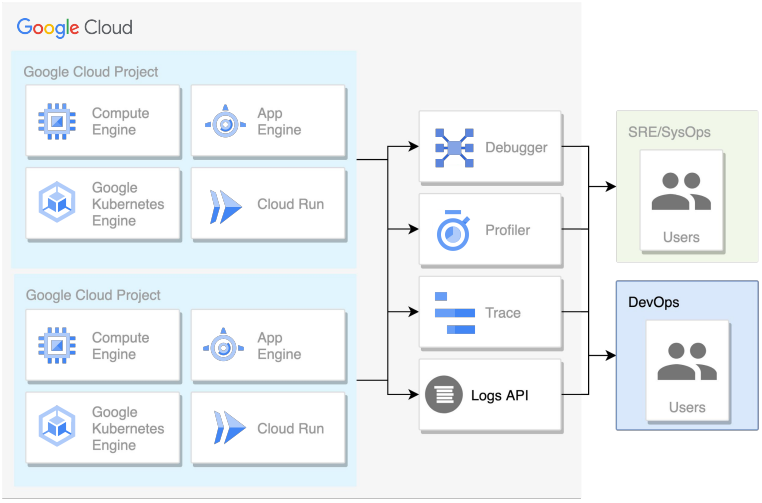
Google Cloud has many products, from Kubernetes, to BigQuery, to Spanner, and they all stream metrics and logs into Google's Cloud Logging and Cloud Monitoring components.

Application and infrastructure observability



Google Cloud has many products, from Kubernetes, to BigQuery, to Spanner, and they all stream metrics and logs into Google's Cloud Logging and Cloud Monitoring components.

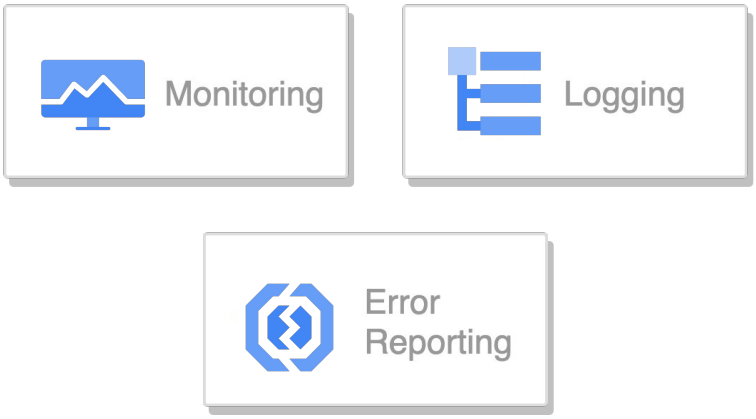
Application performance management tools



In addition to raw monitoring and logging, Google Cloud also helps SysOps/SRE and DevOps personnel analyze and improve application performance. Take, as an example, a containerized HTTP based service running inside the fully managed version of Cloud Run.

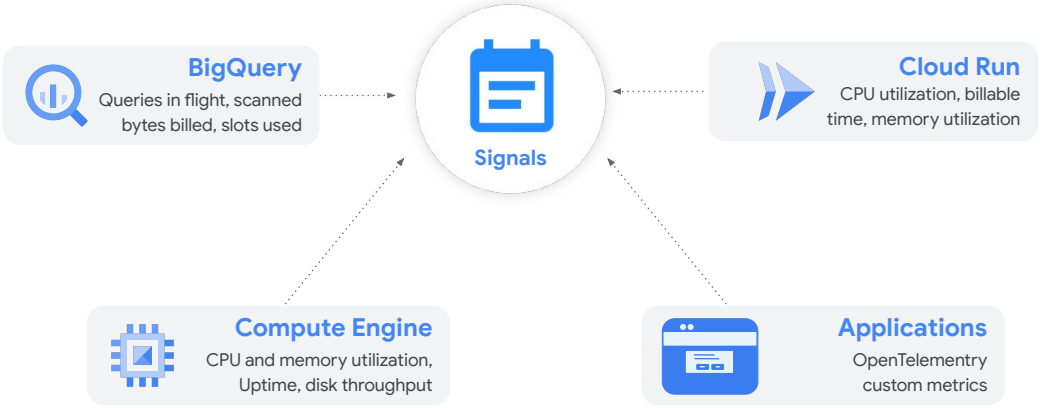
Debugger would allow the inspection of the service's code state without stopping or degrading its performance. It helps answer the question, "What was happening in the code when this particular line executed." Similarly, Profiler can be used to examine CPU and memory utilization to help spot bottlenecks and to improve algorithmic performance. Trace is all about analyzing latency in a multi-layer, microservice application. And the Logs API can be used by developers to write directly to Google Cloud logs.

Operations-based tools



Let's start with the products that tend to be of interest for the operations folk: Monitoring, Logging, and Error Reporting.

Monitoring sources



When DevOps personnel think about tracking exactly what's happening inside Google Cloud projects, a lot of times, the first thing to come to mind is Monitoring. It's mentioned first on the documentation homepage, just like the first product in the operations section of the Google Cloud navigation menu.

As we stated previously, monitoring starts with signal data. Metrics take measurements, and use math to align those measurements over time. Think taking raw CPU usage measurement values and averaging them to produce a single value per minute.

When the data scientists are running massively scalable queries in BigQuery, knowing how many queries are currently in flight, how many bytes have been scanned and added to the bill, and data slot usage patterns, all will be important.

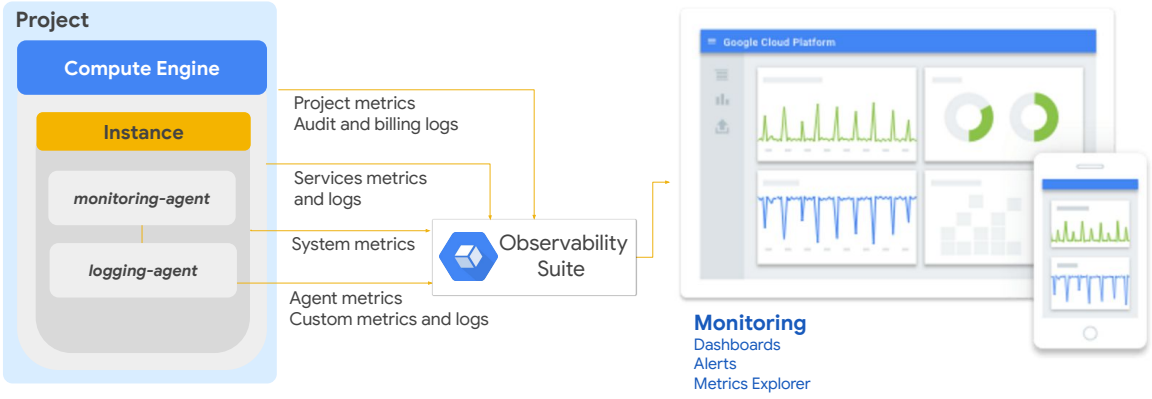
It could also be critical to DevOps teams running containerized applications in Cloud Run to know CPU and memory utilization, and app bill time.

And if those same DevOps teams want to augment the signal metrics coming out of their custom application wherever it's running, they could use the open-source OpenTelementry and create their own metrics.

Workloads on Compute Engine will benefit from CPU and memory utilization data, along with uptime, disk throughput, and scores of others.

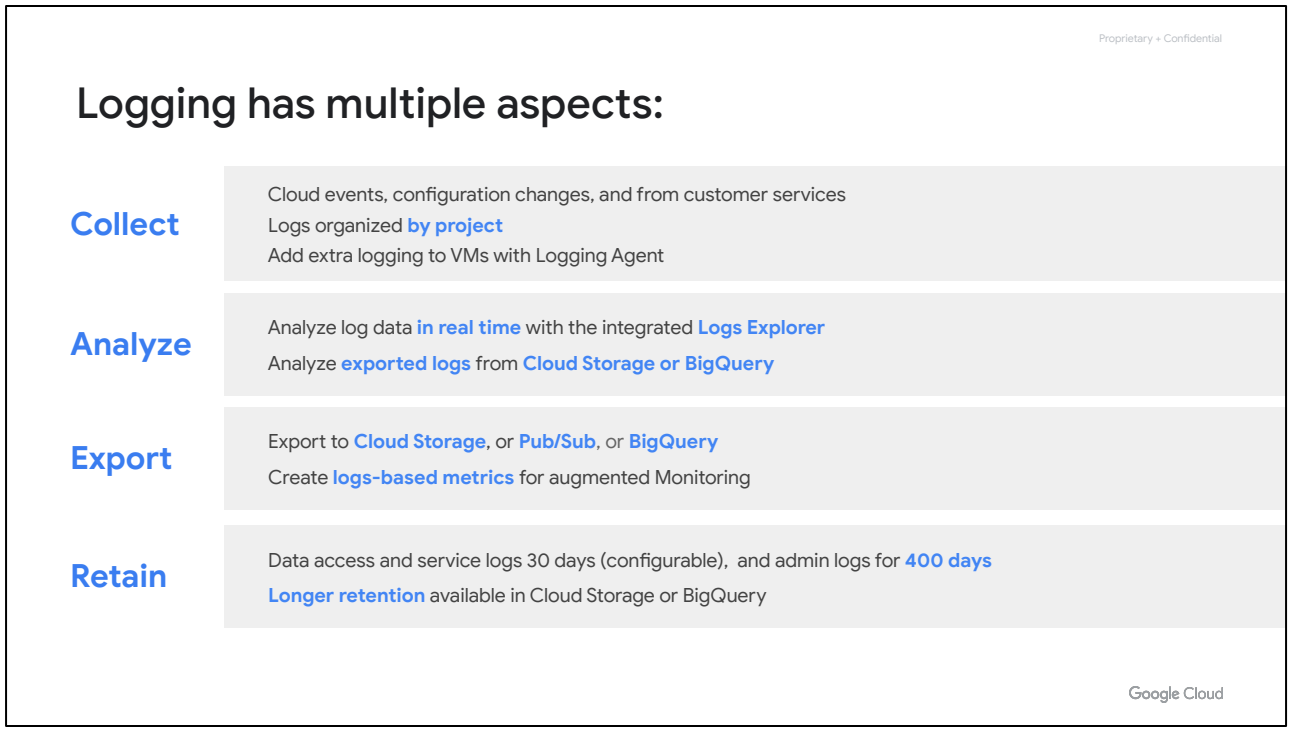
Google Cloud, by default, collects more than a thousand different streams of metric data, which can be incorporated into dashboards, alerts, and a number of other key tools.

Resource monitoring



Here, we see a project running a Compute Engine VM instance with the logging and monitoring agents installed.

Cloud Monitoring provides visibility into the performance, uptime, and overall health of cloud-powered applications. It collects metrics, events, and metadata from projects, logs, services, systems, agents, custom code, and various common application components, including Cassandra, Nginx, Apache Web Server, Elasticsearch, and many others. Monitoring ingests that data and generates insights via dashboards, Metrics Explorer charts, and automated alerts.



Google's Cloud Logging allows users to collect, store, search, analyze, monitor, and alert on log entries and events. Automated logging is integrated into Google Cloud products like App Engine, Cloud Run, Compute Engine VMs running the logging agent, and GKE.

Most log analysis is going to start with the Google Cloud integrated Logs Explorer. Logging entries can also be exported to several destinations for alternative or further analysis. Pub/Sub messages can be analyzed in near-real time using custom code or stream processing technologies like Dataflow. BigQuery allows analysts to examine logging data through SQL queries. And archived log files in Cloud Storage can be analyzed with several tools and techniques.

Export log data as files to Google Cloud Storage, as messages through Pub/Sub, or into BigQuery tables. Logs-based metrics may be created and integrated into Cloud Monitoring dashboards, alerts, and service SLOs.

Default log retention in Cloud Logging depends on the log type. Data access logs are retained by default for 30 days, but this is configurable up to a max of 3650 days. Admin logs are stored by default for 400 days. Export logs to Google Cloud Storage or BigQuery to extend retention.

Available logs

Cloud audit logs

- "Who did what, where?"
- Admin activity
- Data access
- System event
- Access transparency

Agent logs

- Fluentd agent
- Common third-party applications
- System software

Network logs

- VPC flow
- Firewall rules
- NAT gateway
- Load balancer

Service/app logs

- Standard out/error
- Created with API

The Google Cloud platform logs visible to you in Cloud Logging vary, depending on which Google Cloud resources you're using in your Google Cloud project or organization. Four key log categories are audit logs, agent logs, network logs, and service logs.

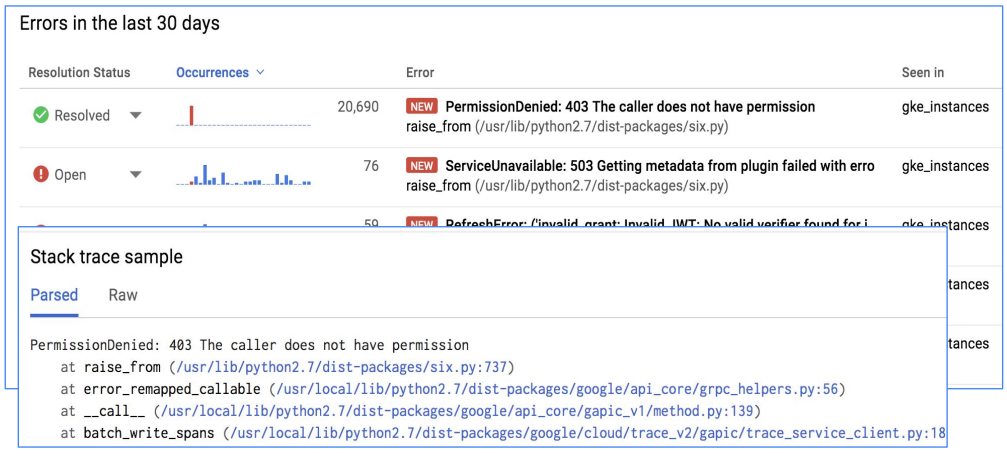
Cloud Audit Logs help answer the question, "Who did what, where, and when?" Admin activity tracks configuration changes. Data access tracks calls that read the configuration or metadata of resources, as well as user-driven calls that create, modify, or read user-provided resource data. System events are non-human Google Cloud administrative actions that change the configuration of resources. Access Transparency provides you with logs that capture the actions Google personnel take when accessing your content.

Agent logs use a Google-customized and packaged Fluentd agent that can be installed on any AWS or Google Cloud VM to ingest log data from Google Cloud instances (for example, Compute Engine, Managed VMs, or Containers), as well as AWS EC2 instances.

Network logs provide both network and security operations with in-depth network service telemetry. VPC Flow Logs record samples of VPC network flow and can be used for network monitoring, forensics, real-time security analysis, and expense optimization. *Firewall Rules Logging* allows you to audit, verify, and analyze the effects of your firewall rules. NAT Gateway logs capture information on NAT network connections and errors.

Service logs provide access to logs created by developers deploying code to Google Cloud. For example, if they build a container using NodeJS and deploy it to Cloud Run, any logging to Standard Out or Standard Error will automatically be sent to Cloud Logging for easy, centralized viewing.

Error reporting



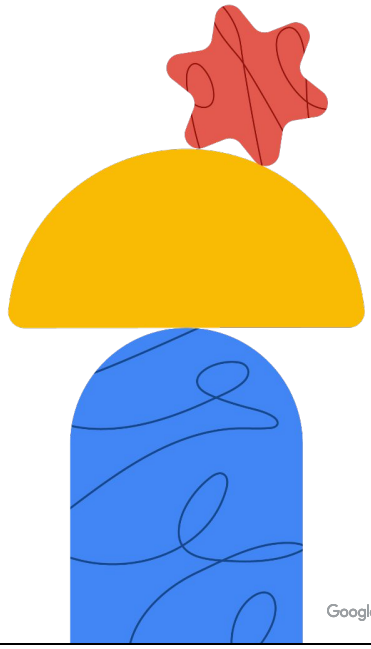
Error Reporting counts, analyzes, and aggregates the crashes in your running cloud services. Crashes in most modern languages are Exception which are not caught and handled by the code itself. Its management interface displays the results with sorting and filtering capabilities. A dedicated view shows the error details: time chart, occurrences, affected user count, first- and last-seen dates, and a cleaned exception stack trace. You can also create alerts to receive notifications on new errors.

Application performance management tools

Profiler

Trace

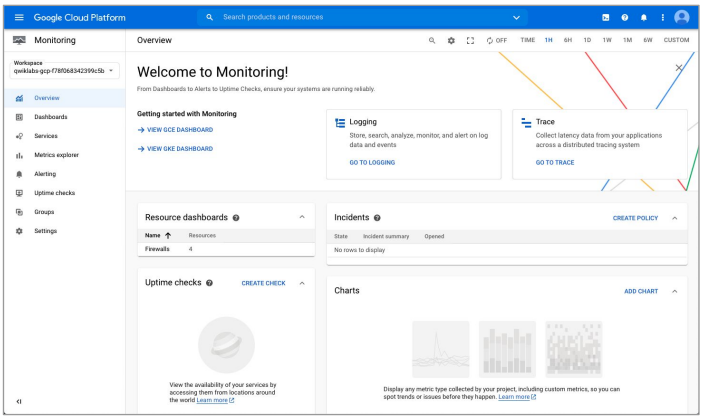
Monitoring critical systems



Let's spend a little time talking about how Google Cloud helps you monitor critical systems.

Monitoring is configured via Workspaces

- Single pane of glass
- Cross-project visibility
- Monitor resources in Google Cloud and AWS



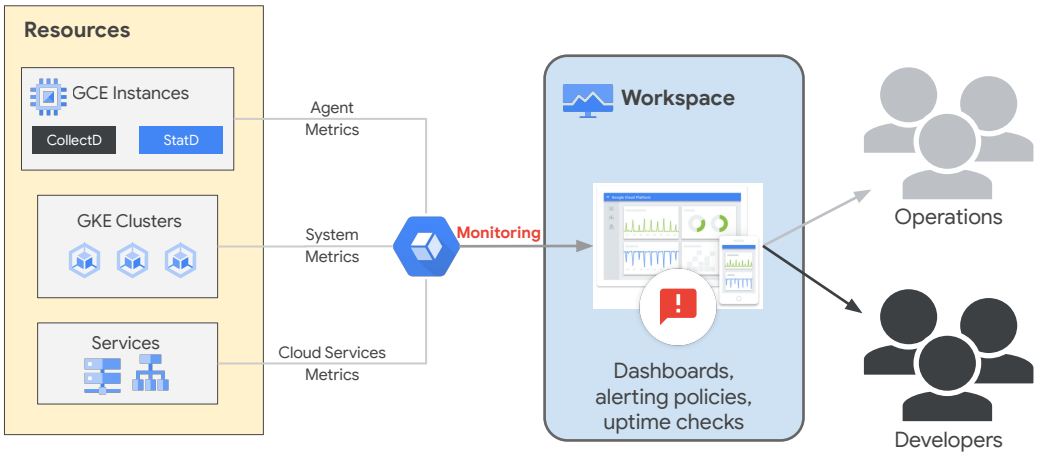
Google Cloud Monitoring uses Workspaces to organize monitoring information. A Workspace is a tool for monitoring resources contained in one or more Google Cloud projects.

It offers a unified view, or single pane of glass, through which those resources can be watched.

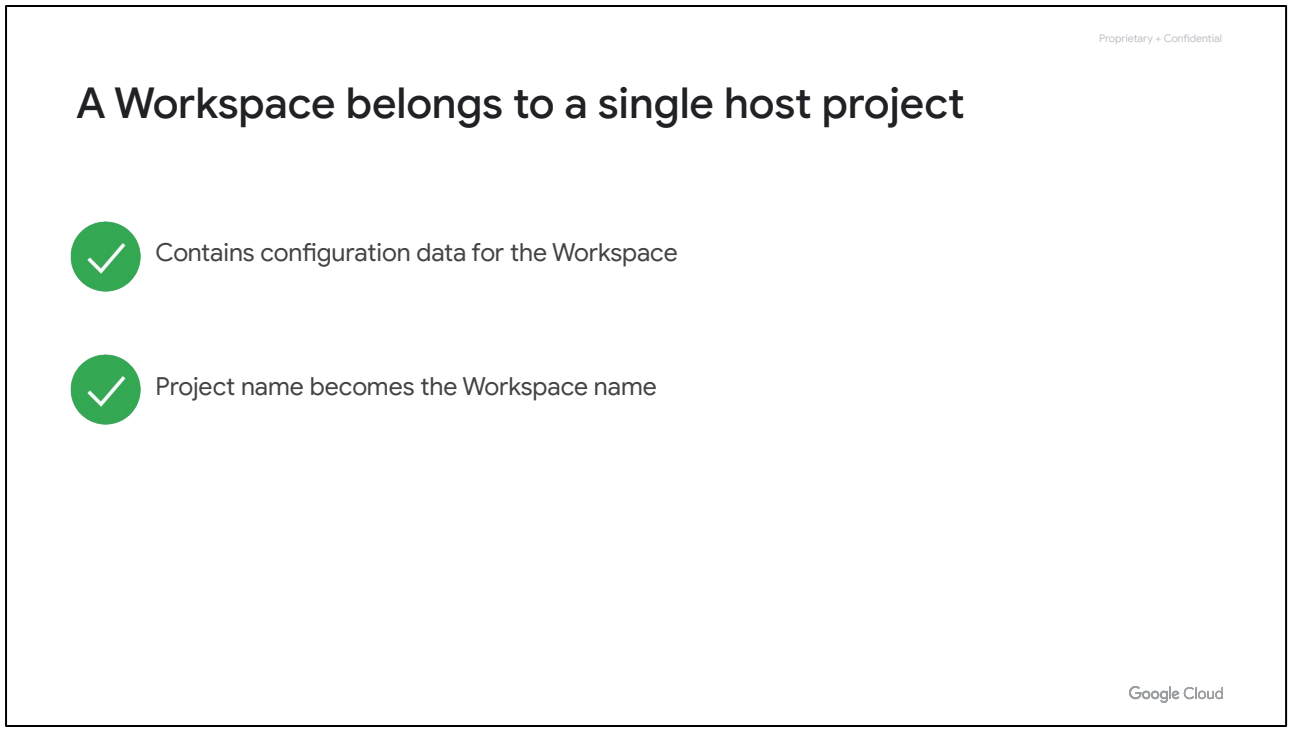
With the ability to monitor resources in the current project, and in up to 100 other projects, monitoring workspaces offer excellent cross-project visibility.

The monitored resources may be part of Google Cloud or AWS.

Organize your monitoring efforts with Workspaces



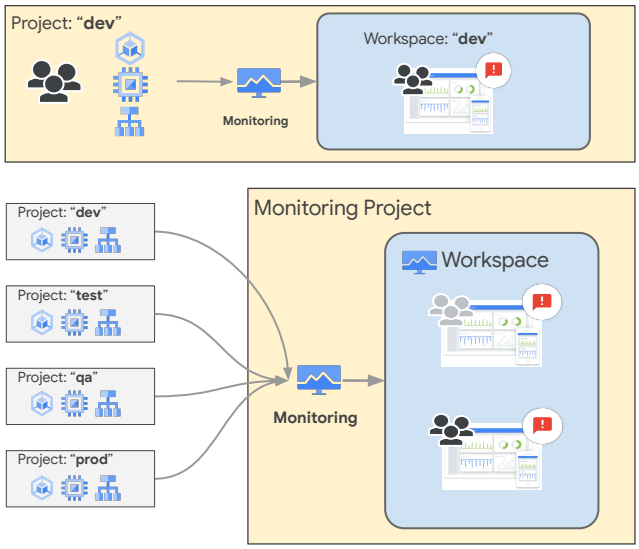
Monitoring Workspaces help organize your monitoring efforts. They serve as central, secured access hubs for monitoring information, dashboards, alerting policies, and uptime checks. This information is made available, IAM permitting, to both operations and developer personnel.



A Workspace belongs to a single host project. The host project stores all of the configuration content for dashboards, alerting policies, uptime checks, notification channels, and group definitions that you configure. If you delete the host project, you also delete the Workspace.

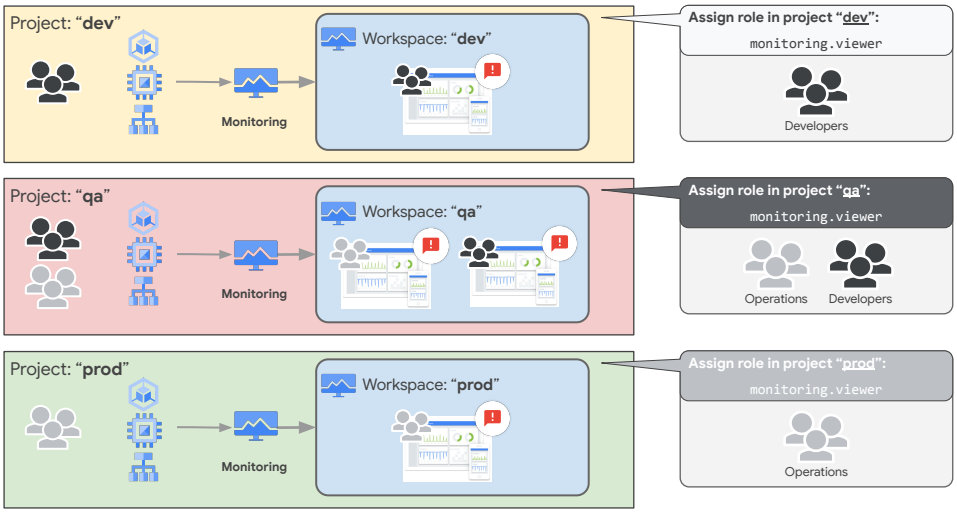
The name of the Workspace is set to the name of the host project. This isn't configurable.

Two options for monitoring Workspace architectures



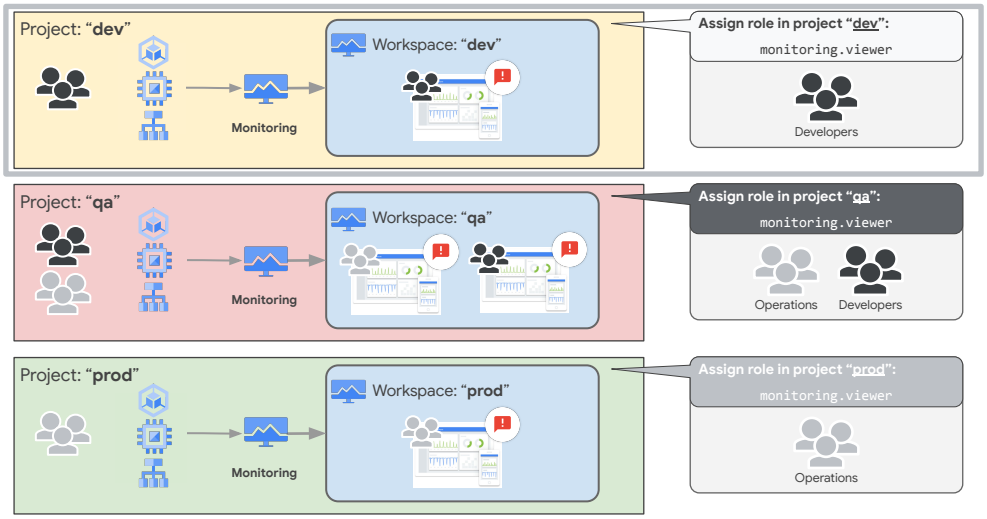
There are only two options on how a project with resources is monitored. Either the project contains the resources and all the monitoring, or a project contains resources and it's monitored by an externally configured "monitoring" project.

Monitor by project for maximum isolation



Strategy C: Every project is monitored locally, in that project.

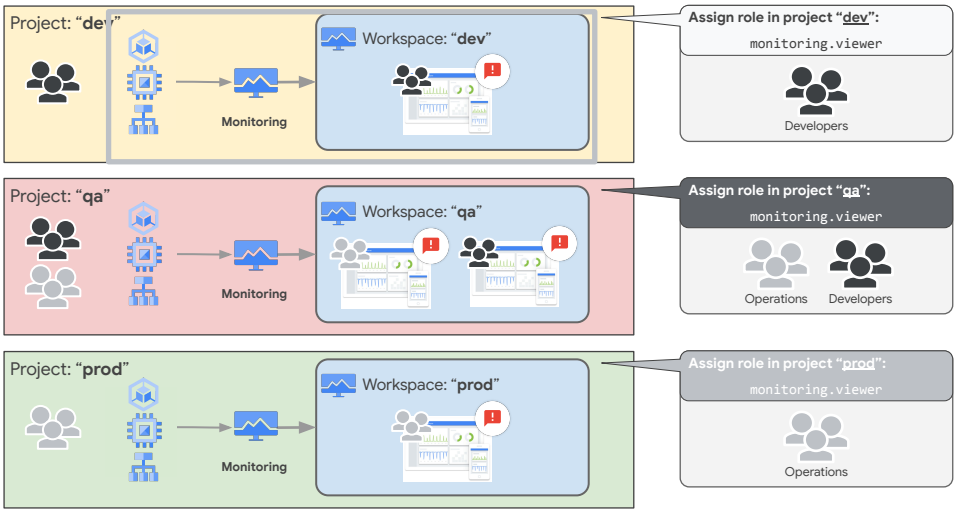
Monitor by project for maximum isolation



Advantages:

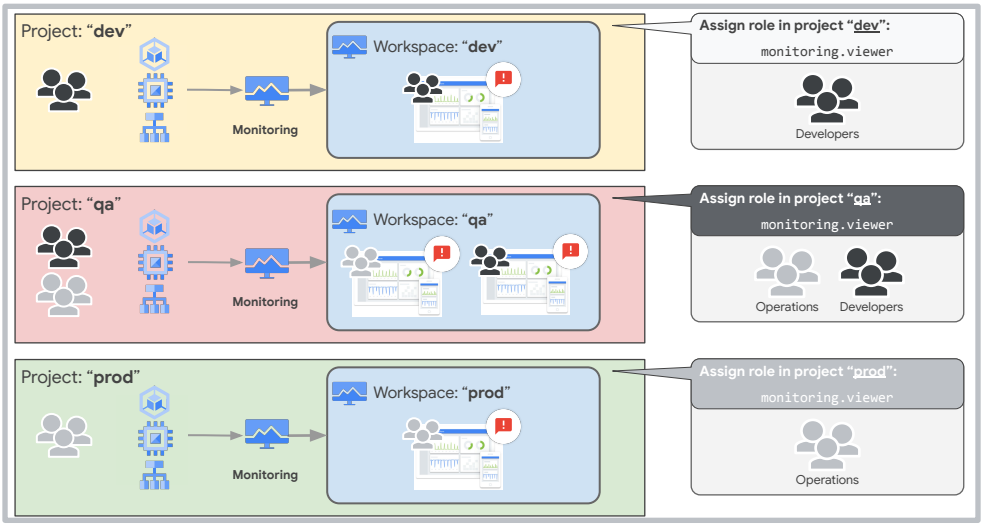
- Clear and obvious separation for each project. If the project contains dev-related resources, it's easy to provide access to the dev personnel.

Monitor by project for maximum isolation



Project resources and monitoring resources all in the same place.

Monitor by project for maximum isolation



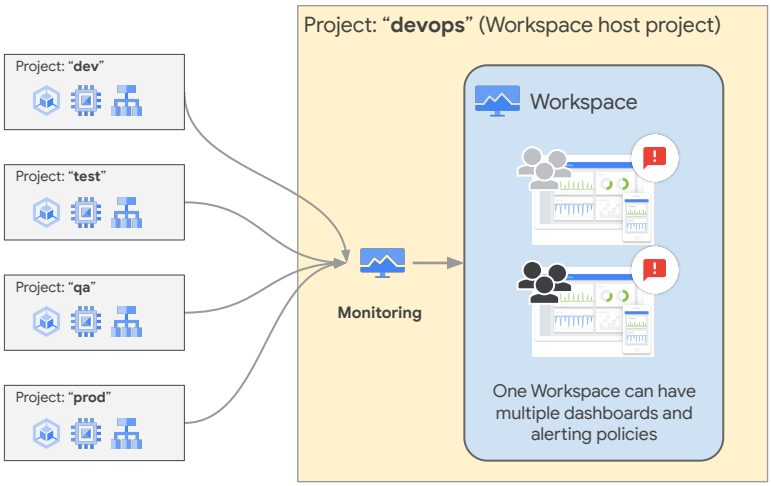
Disadvantages:

- If the application is larger than a single project, then you will be looking at a small slice of a bigger picture, and bringing that full picture into focus might be much harder to do.

One Workspace can monitor multiple projects

Since it's possible for one Workspace to monitor multiple projects, but a project can be monitored from only a single Workspace, you will have to decide which Workspace-to-project relationship will work best for your organizational culture, and this particular project.

One Workspace can monitor multiple projects

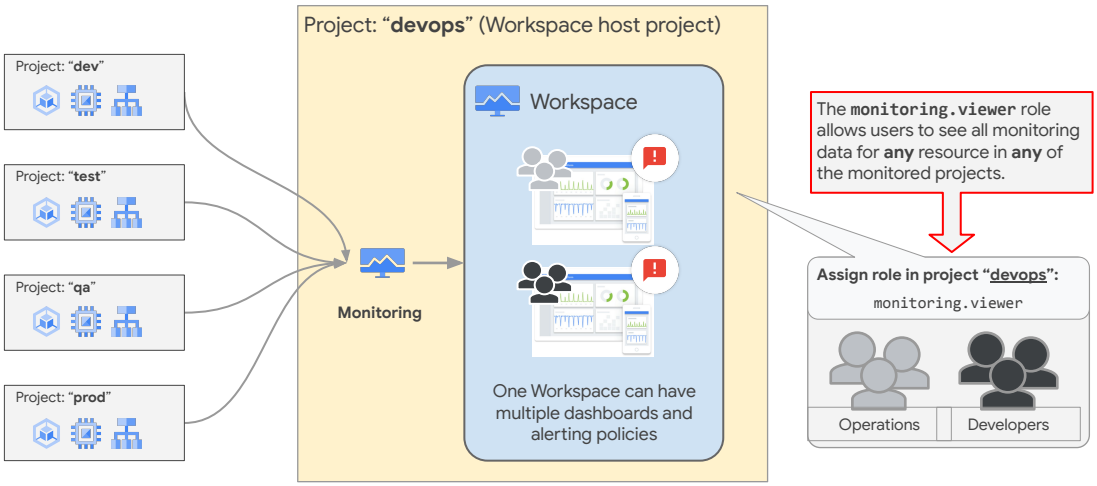


Strategy A: Single monitoring Workspace for large units of projects, probably an application or application part.

Advantages:

- Single pane of glass that provides visibility into the entire group of related projects.
- Can compare non-prod and prod environments easily.

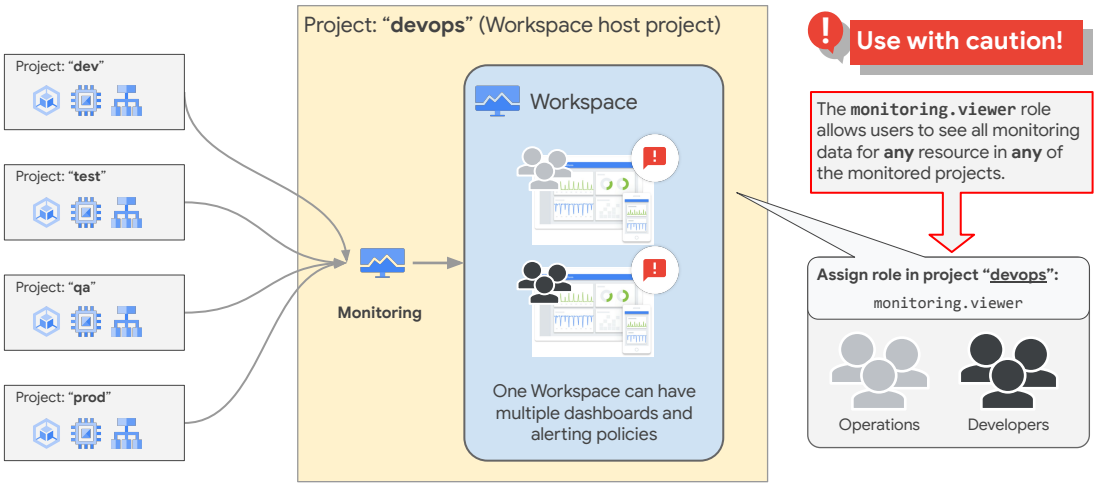
One Workspace can monitor multiple projects



Disadvantages:

- Anyone with IAM permissions to access Monitoring will be able to see metrics for all environments.
- Monitoring in prod is usually done by different teams; this approach wouldn't allow that delineation.

One Workspace can monitor multiple projects

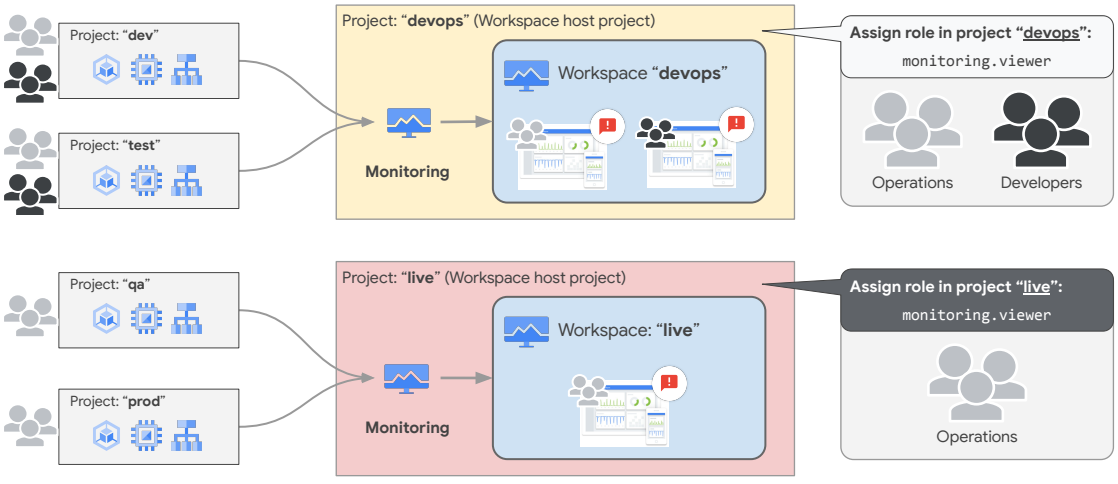


Although the metric data and log entries remain in the individual projects, any user who has been granted the role Monitoring Viewer (roles/monitoring.viewer) will have access to the dashboards and have access to all data by default. This means that a role assigned to one person on one project applies equally to all projects monitored by that Workspace.

Logical groupings can be very effective

To give people different roles per project, and to better control visibility to data, consider smaller, more selective, monitoring Workspaces.

Logical groupings can be very effective



Strategy B: Prod and Non-Prod monitoring Workspaces.

Advantages:

- Clear delineations between production and the other environments.
- Lowers the maintenance burden of too many monitoring Workspaces (such as in Strategy C).
- Logical boundaries don't have to be production, non-production. This approach of small groups of projects being monitored centrally can apply to many different Google Cloud architectures.

Disadvantages:

- Have to be careful of the monitored project groupings.
- This approach still provides multi-project access to monitoring data.

IAM Roles control user access to Workspaces

To initially create the Monitoring Workspace, a user will need the Monitoring Editor or Monitoring Admin role in the Workspace host project.

Role Name	Description
Monitoring Viewer	Gives you read-only access to the Monitoring console and API
Monitoring Editor	Gives you read-write access to the Monitoring console and API, and lets you write monitoring data to a Workspace
Monitoring Admin	Gives you full access to all Monitoring features

There are a number of IAM security roles related to monitoring. The big three are viewer, editor, and admin.

To create the monitoring Workspace initially, a user will need the Monitoring Editor or Admin role in the the Workspace's host project.

Past that, a Monitoring Viewer can get read-only access to the Monitoring console and API.

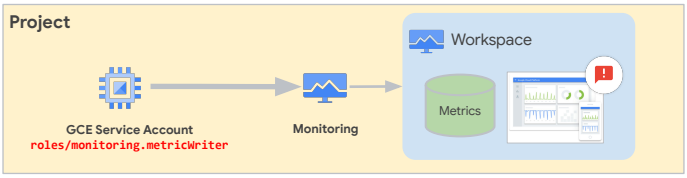
Monitoring Editor has read-write access to the Monitoring console and APIs and can write monitoring data and configurations into the Workspace.

Monitoring Admin has full access to, and control over, all monitoring resources.

Past these big three roles, monitoring roles exist to provide and limit access to alert policies, dashboards, notification channels, service monitoring, and uptime checks. Check the documentation for more information: https://cloud.google.com/monitoring/access-control.

Services may need permission to add metric data

For example, the service account of a GCE instance with the monitoring agent installed - grant the service account the Monitoring Metric Writer role in the Workspace host project.



Monitoring Metric Writer

Permits writing monitoring data to a Workspace.

This does not permit read access to the Monitoring console.

Typically this permission is used by service accounts.

Another critical security role is *metricWriter*. Services may need permission to add metric data to the monitoring Workspace.

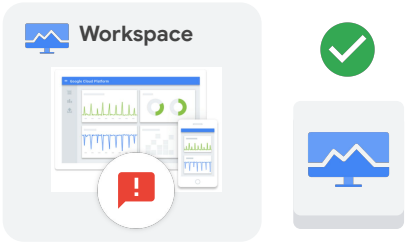
For example, take a Google Compute Engine VM running an agent that needs to stream metrics into the monitoring Workspace.

To allow it the write access it needs, grant the VM's service account the Monitoring Metric Writer role in the Workspace host project.

Monitoring Metric Writer permits writing monitoring data to a Workspace. This does not permit read access to the Monitoring console. Typically, this permission is used by service accounts, as in this example.

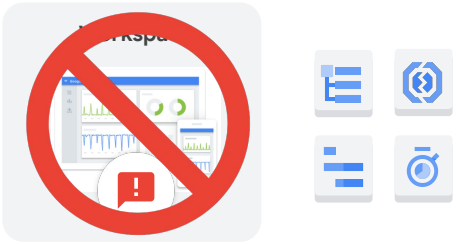
Remember, workspaces only affect monitoring

Only the Monitoring system relies upon workspaces.



The other tools in this course:

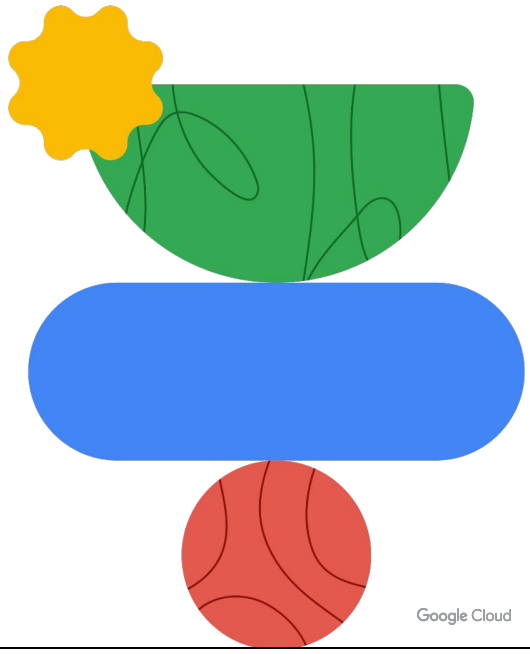
- Are configured on a per-project basis.
- Have their own Cloud IAM roles.



Remember, Monitoring Workspaces only affect and control Google Cloud resources related to monitoring.

Other tools covered in this course, such as Logging, Error Reporting, and the Application Performance Management (APM) tools, are strictly project-based and do not rely upon the configuration of the Monitoring Workspaces or the monitoring IAM roles.

Alerting policies

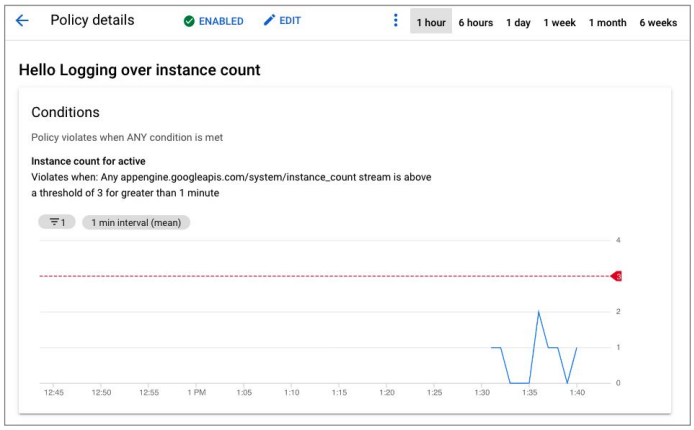


Alerting gives timely awareness to problems in your cloud applications so you can resolve the problems quickly.

Use alerting policies to define alerts

An alerting policy has:

- A name
- One or more conditions
- Notifications
- Documentation



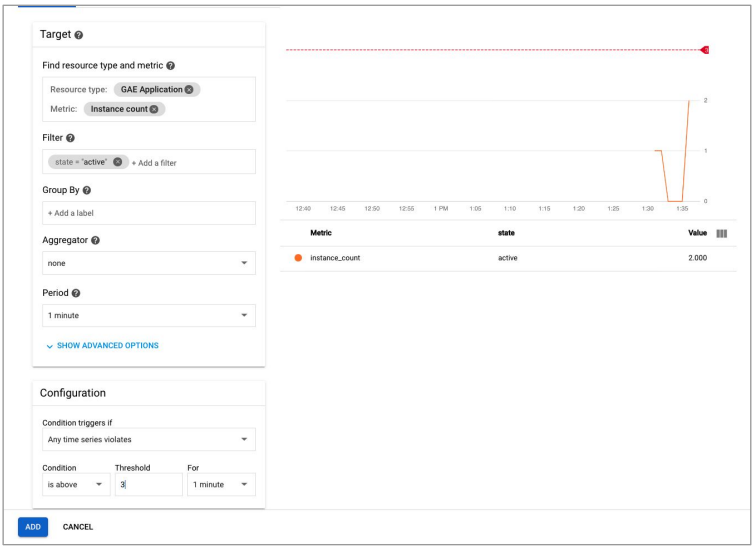
Google Cloud defines alerts using alerting policies.

An alerting policy has:

- A name
- One or more alert conditions
- Notifications
- And documentation

For the name, use something descriptive so you can recognize alerts after the fact. Organizational naming conventions can be a great help.

Conditions: What's watched and when to alert

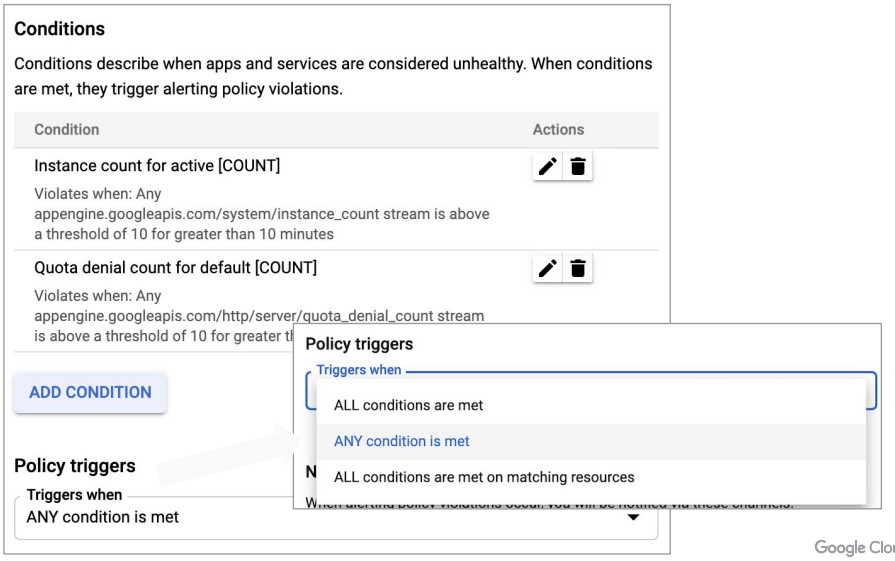


The alert condition is where you'll be spending the most alerting policy time and making the most decisions. This is where you decide what's being monitored and under what condition an alert should be generated. Notice how the UI combines the heart of the Metrics Explorer with a configuration condition.

You start with a target resource and metric you want the alert to monitor. You can filter, group by, and aggregate to the exact measure you require.

Then the yes-no decision logic for triggering the alert notification is configured. It includes the trigger condition, threshold, and duration.

Use multiple conditions

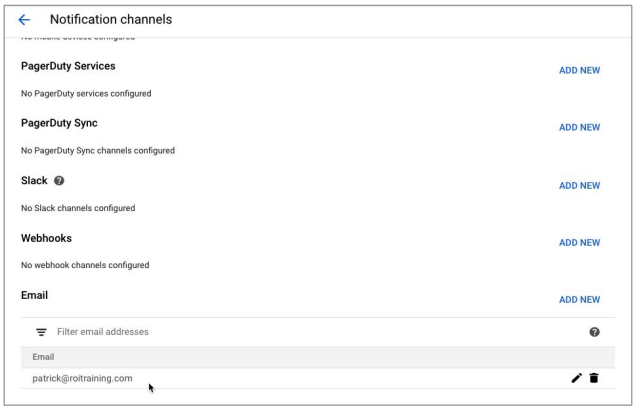


To try to maximize both precision and recall within a single alert; you can create multiple conditions. The policy trigger is used to determine how more than one trigger will relate to one another and to the alert triggering itself.

Select notification channels

Supported notification channels include:

- Email
- SMS
- Slack
- Google Cloud Mobile app
- PagerDuty
- Webhooks
- Pub/Sub



The notification channel, or channels, decides how the alert is sent to the recipient.

Email alerts are easy and informative, but they can become notification spam if you aren't careful.

SMS is a great option for fast notifications, but choose the recipient carefully.

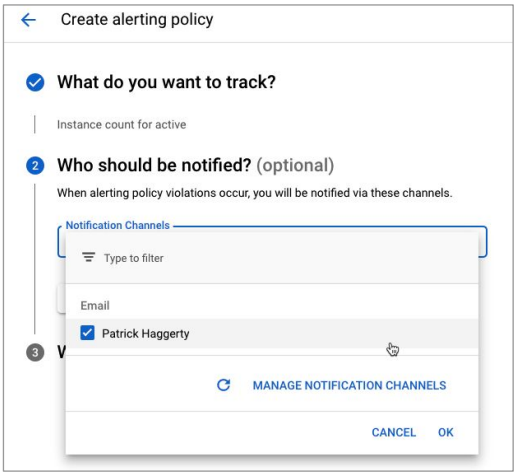
Slack is very popular in support circles.

The Google Cloud mobile app is a valid option.

PagerDuty is a third party on-call management and incident response service.

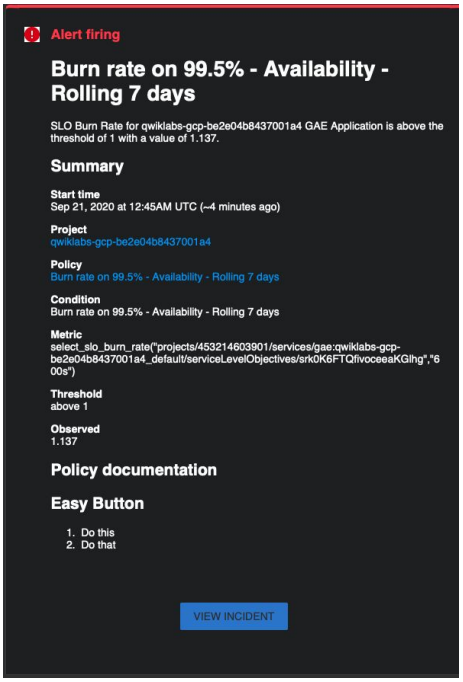
Webhooks and Pub/Sub are excellent options when alerting to external systems or code.

Zero to many notification channels



An alert may have zero to many notification options selected, and they each can be of a different type.

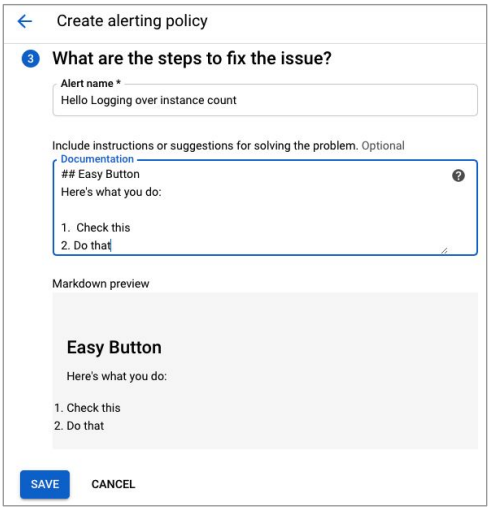
Email alert example



Here, you see an alert notification sent out through an email. Notice how a lot of the details about exactly what went wrong are automatically included in the email body. The bottom documentation section can also be used to augment the provided information.

Include documentation for added clarity

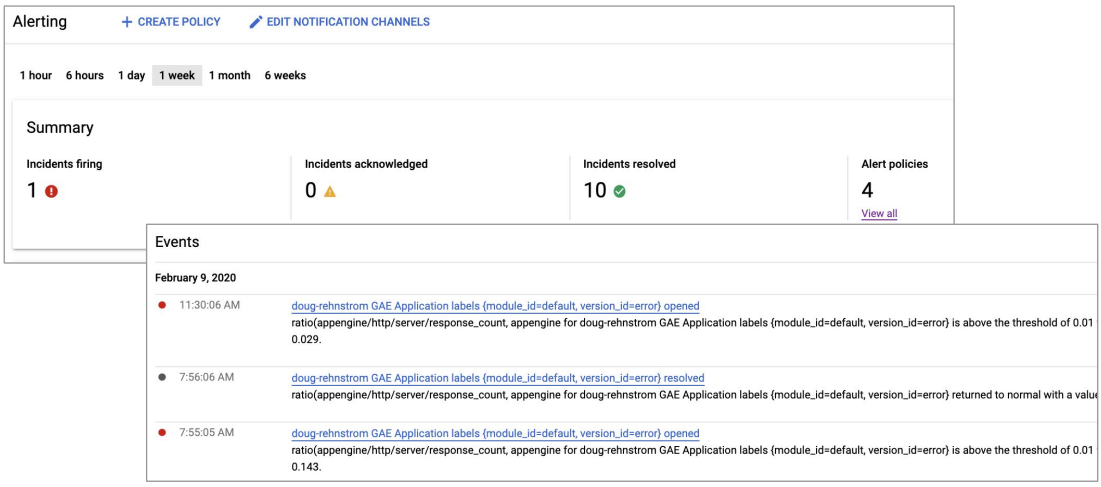
- Make it easy for the team to understand what is wrong.
- Use markdown to format messages.
- This should be the easy button for the fix, if there is one.



The documentation option is designed to give the alert recipient additional information they might find helpful. The default alert content will already contain information about which alert is failing and why, so think of this more like an easy button. If there's a standard solution to this particular alert, adding a reference to it here might be a good example of proper documentation inclusion.

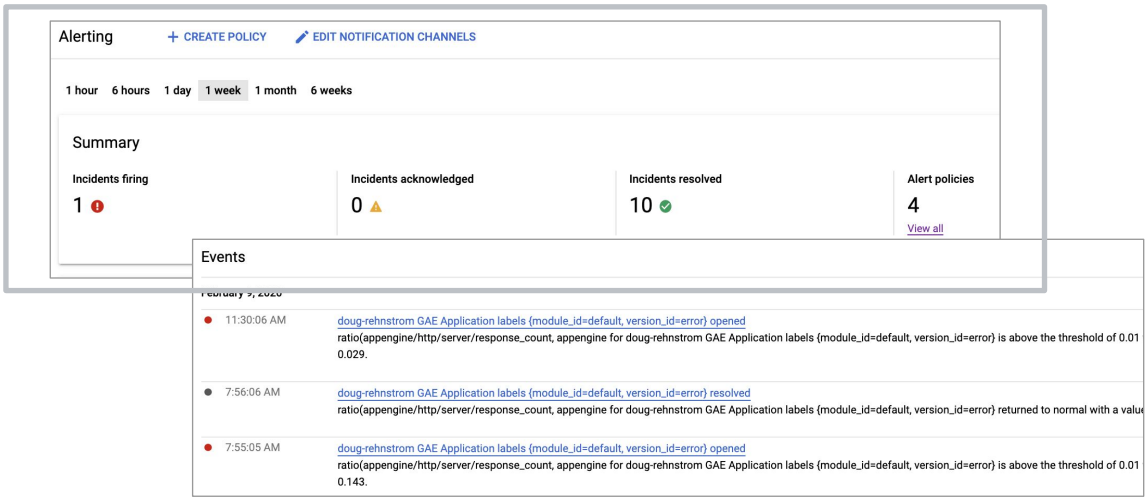
Then again, if it was that easy, automate it!

Alerting UI summarizes incidents and events



When one or more alert policies have been created, the Alerting UI provides a summary of incidents and alerting events. An event occurs when the conditions for an alerting policy are met. When an event occurs, Cloud Monitoring opens an incident.

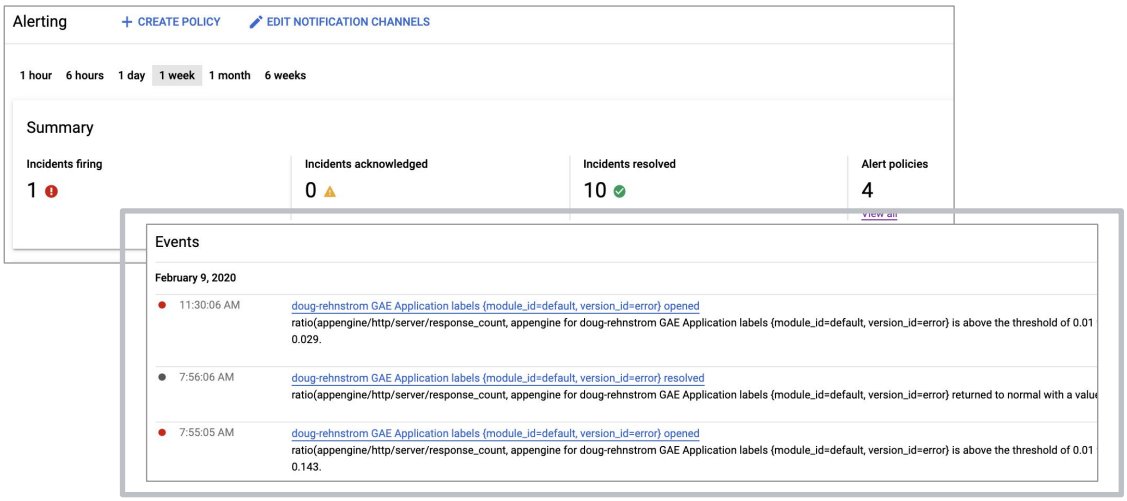
Alerting UI summarizes incidents and events



In the **Alerting** window, the **Summary** pane lists the number of incidents, and the **Incidents** pane displays the 10 most recent incidents. Each incident is in one of three states:

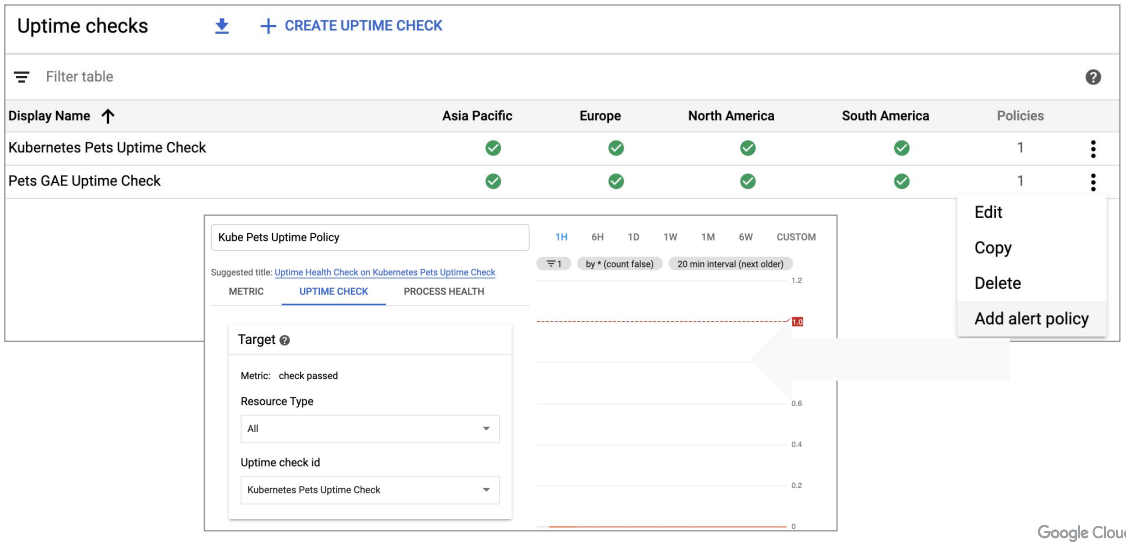
- Open incidents. If an incident is open, the alerting policy's set of conditions is currently being met, or there is no data to indicate that the condition is no longer met. This usually indicates a new or unhandled alert.
- Acknowledged incidents. A tech spots a new open alert, but before they start to investigate, they mark it as acknowledged as a signal to others that someone is dealing with the issue.
- Closed incidents. If an incident is closed, the alert policy conditions are no longer being met. An incident is listed as closed if there is no data to indicate whether the condition still exists and the incident has expired.

Alerting UI summarizes incidents and events



The **Events** pane of the **Alerting** dashboard displays the most recent events and includes a graphical indicator of the alert status, a link to event details, a quick description, and a timestamp.

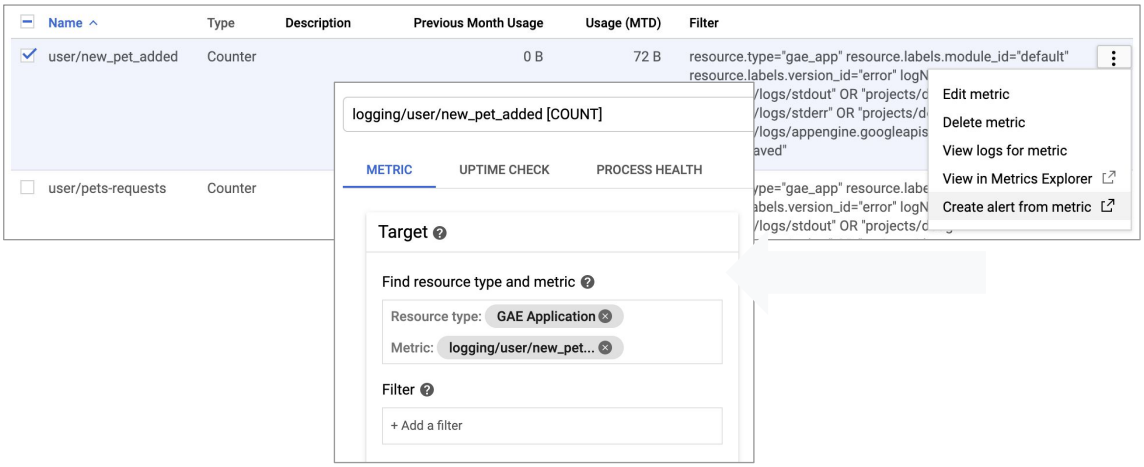
Attach alerts to uptime checks



An uptime check is a request sent to an externally accessible site or service to see if it responds, or is up. You can use uptime checks to determine the availability and latency of a VM instance, an App Engine service, a URL, or an AWS load balancer.

You can monitor the availability of a resource by creating an alerting policy that creates an incident if the uptime check fails. You also have the option to observe the results of uptime checks in the Monitoring uptime-check dashboards.

Attach alerts to logs-based metrics

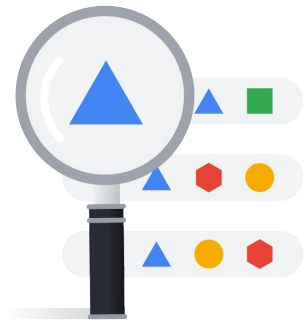


Logs-based metrics are Cloud Monitoring metrics based on the content of log entries. For example, the metrics can record the number of log entries containing particular messages, or they can extract latency information reported in log entries. You can use logs-based metrics in Cloud Monitoring charts and alerting policies.

As we covered earlier in this module, an alerting policy describes a set of conditions that you want to monitor. When you create an alerting policy, you must also specify its conditions: what is monitored and when to trigger an alert. The logs-based metric serves as the basis for an alerting condition.

Resource groups can monitor multiple resources

- Trigger based on the group instead of on individual resources.
- Groups can contain subgroups up to six levels deep.
- Resources can be members of more than one group.
  - Max of 500 groups per monitoring workspace



Groups provide a mechanism for alerting on the behavior of a set of resources, rather than on individual resources. For example, you can create an alerting policy that is triggered if some number of resources in the group violates a particular condition (for example, CPU load), rather than having each resource inform you of violations individually.

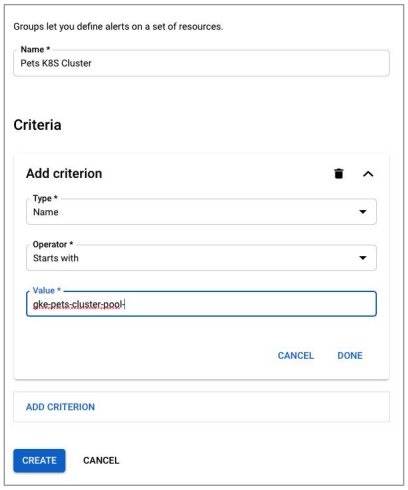
Groups can contain subgroups, up to six levels deep. One application for groups and subgroups is the management of physical or logical topologies. For example, with groups, you can separate your monitoring of production resources from your monitoring of test or development resources. You can also create subgroups to monitor your production resources by zone.

Resources can belong to multiple groups, and a given monitoring workspace can have up to 500 groups.

Use multiple criteria to create resource groups

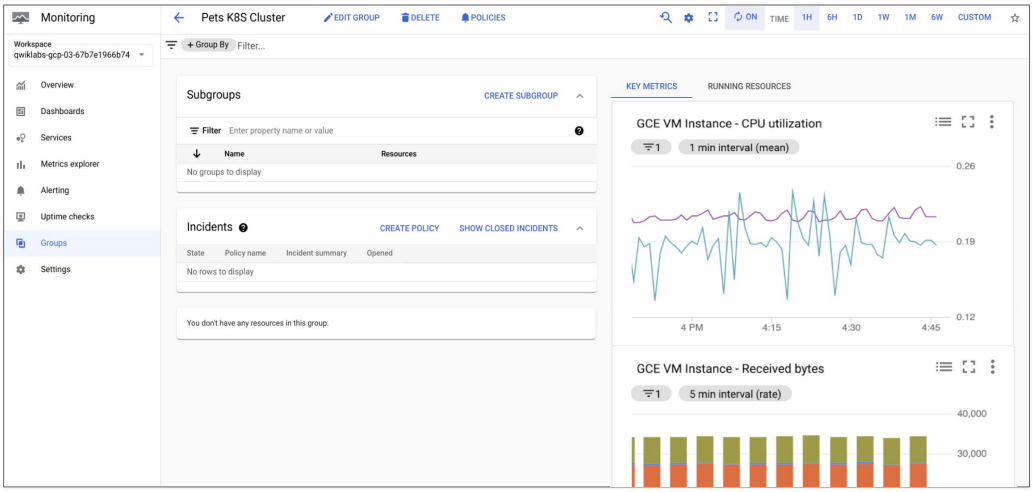
Criteria can include:

- Resource name
- Resource type
- Tags and labels
- Security groups
- Regions
- App Engine apps and services



You define the one-to-many membership criteria for your groups. A resource belongs to a group if the resource meets the membership criteria of the group. Membership criteria can be based on resource name or type, network tag, resource label, security group, region, or App Engine app or service. Resources can belong to multiple groups.

Monitor all resources in a group together



After the group is created, all the resources in the group can be monitored together as a unit.

You have now completed the Networking in Google Cloud module.