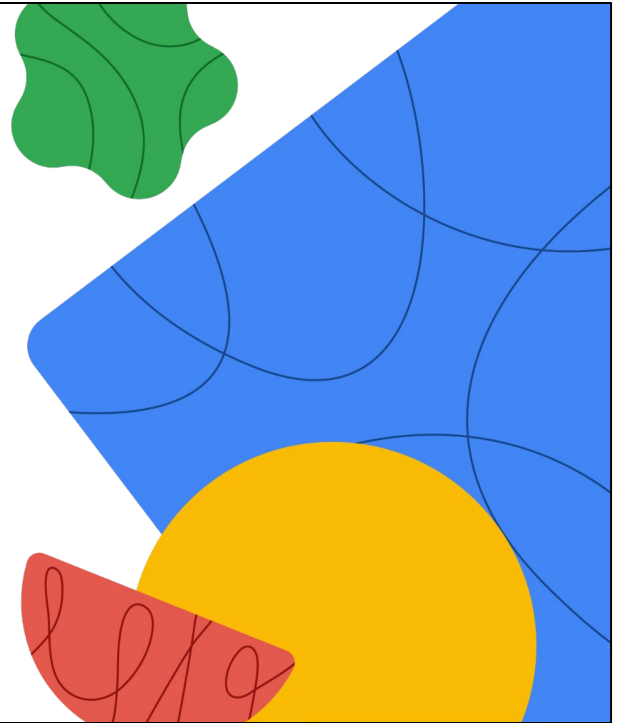




Network in Google Cloud

Network Topologies



Welcome to the Network Topologies module.



Today's agenda



01 Hub-and-spoke topology

02 Lab: Implement a Hub-and-Spoke Network Using Network Connectivity Center

03 Other topologies

04 Getting topology data

05 Best practices

06 Quiz

Google Cloud

This module introduces you to network topologies. We'll start with the hub and spoke model, exploring its structure and applications. Additionally, we'll cover alternative topologies like mesh, mirrored, and gated. Through a hands-on lab, you will explore how to implement a hub and spoke using Network Connectivity Center.

Let's start with a use case.

Needed: a simple topology for centralized control

- ✓ Nur, a network engineer at Cymbal Corporation, faces challenges managing a growing network.
- ✓ Expanding remote offices and cloud-based applications require a scalable and manageable solution.
- ✓ Nur seeks a simple network topology for centralized control and efficient data management.

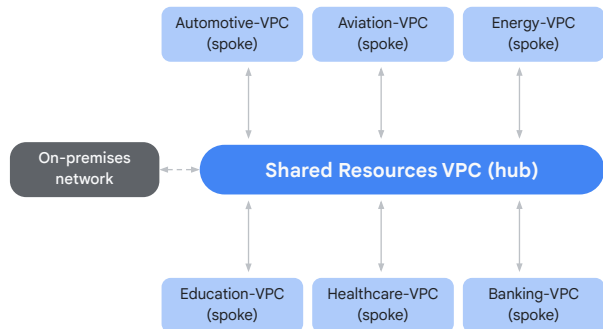


Google Cloud

Nur, a network engineer at Cymbal Corporation, is tasked with managing the company's expanding network infrastructure, which encompasses a growing number of remote offices and cloud-based applications. To ensure centralized control and efficient data management, Nur wants a simple network topology that can be centrally managed with simple network administration.

Solution: Hub-and-spoke topology

- ✓ Nur chooses a hub-and-spoke topology, where a central hub connects to multiple network devices or spokes.
- ✓ The hub acts as a central point of control for managing and monitoring the entire network.
- ✓ Spoke devices can be diverse, including remote offices, cloud instances, and on-premises data centers.



Google Cloud

Nur decided to use a hub-and-spoke topology. The hub-and-spoke topology is like a star, with the central hub at the center and all the other devices connected to it like spokes on a wheel. Instead of managing each device individually, you can configure and monitor everything from the hub, saving you time and effort.

This topology is a common approach for managing network traffic in Google Cloud environments. One Virtual Private Cloud (VPC) network (the hub) acts as a transit point for the other VPC networks (the spokes). Additionally, spoke devices can be diverse, including remote offices, cloud instances, and on-premises data centers.

Solution: Hub-and-spoke topology

A hub-and-spoke topology features:



A centralized point of control



Simplified network administration



Scalability



Improved security

Google Cloud

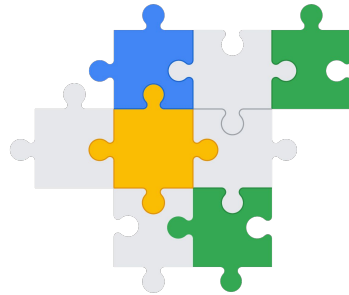
A hub-and-spoke topology features:

- A centralized point of control: Nur can easily configure, monitor, and troubleshoot the entire network from the central hub. This eliminates the need to log in to individual devices for management tasks.
- Simplified network administration. There is no need for complex routing configurations between individual devices. The hub automatically routes traffic between spokes based on predefined rules.
- Scalability: The hub-and-spoke topology can easily accommodate future network growth by adding more spokes. New devices can be quickly connected to the hub without impacting existing network configurations.
- Improved security: Centralized security policies can be enforced at the hub, enhancing overall network security. This simplifies security management and ensures consistent protection across all connected devices.

Possible implementations

There are multiple ways to implement a hub-and-spoke topology, for example:

- ✓ VPC Network Peering
- ✓ Cloud VPN
- ✓ Network Connectivity Center



Google Cloud

A hub-and-spoke network topology is a versatile approach for managing network traffic, but there are different ways to implement it. This slide highlights three common methods:

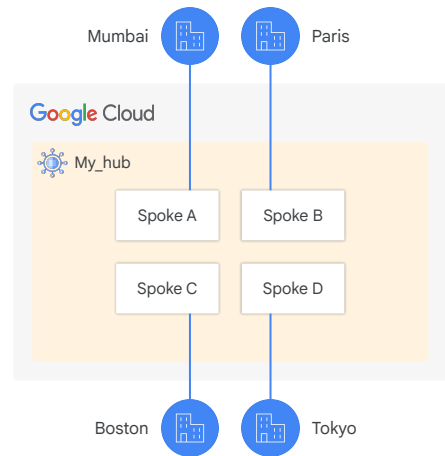
VPC Network Peering allows direct connections between VPCs, enabling efficient communication with low latency.

Cloud VPN creates a secure tunnel over the public internet to connect VPCs across regions or even to on-premises networks. While offering flexibility, Cloud VPN may introduce higher latency and require additional configuration compared to VPC peering.

Network Connectivity Center (NCC) simplifies building and managing complex network topologies, including hub-and-spoke. It provides centralized configuration, automated provisioning, and integrated security features. However, Network Connectivity Center may come with additional costs compared to basic VPC peering or Cloud VPN setups. In this course, we will cover the NCC implementation.

Implement a hub and spoke using Network Connectivity Center

- Spoke types include:
 - A VPC network
 - Hybrid Spoke
 - HA VPN tunnels
 - Cloud Interconnect VLAN attachments
 - Router appliance spokes



Network Connectivity Center simplifies network management by supporting two types of spokes.

VPC spokes connect Virtual Private Clouds, while hybrid spokes connect on-premises networks using HA VPN tunnels or Cloud Interconnect VLAN attachments with router appliance VMs.

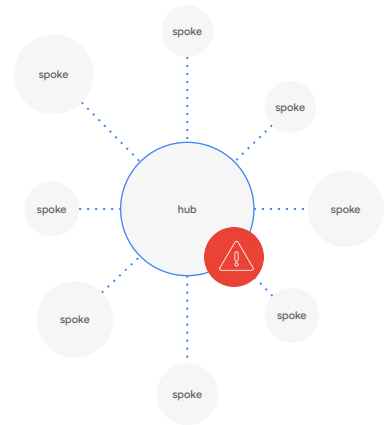
You can configure a router appliance instance by installing an image on a Compute Engine VM. You can use an image provided by a supported Network Connectivity Center partner. You can also use a custom image, such as an image that you created. You can then take advantage of logic specific to the router appliance instance to control connectivity between the spoke and the hub.

Additional considerations

01 Ensure that IP address spaces between the hub, spoke, and on-premises networks don't overlap.

02 IPv6 addressing isn't supported.

03 Privately-used public IP addresses (PUIs) aren't supported.




Google Cloud

Before you create the hub-and-spoke topology using Network Connectivity Center, ensure that IP address spaces between the hub and spoke VPC networks don't overlap.


As with VPC Network Peering, ensure that IP address spaces among the hub, spoke, and on-premises networks don't overlap.

IPv6 addressing isn't supported. Privately-used public IP addresses (PUI) are not supported either. If you need to use either of these two features, consider using VPC Network Peering to create a hub-and-spoke topology.

For more information, refer to the [Considerations](#) section of the [Network Connectivity Center Overview](#) in the Google Cloud documentation.



Today's agenda

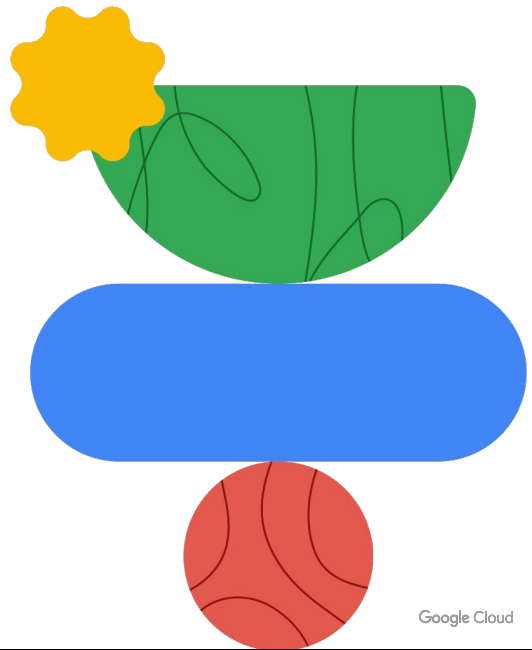


- 01 Hub-and-spoke topology
- 02 [Lab: Implement a Hub-and-Spoke Network Using Network Connectivity Center](#)
- 03 Other topologies
- 04 Getting topology data
- 05 Best practices
- 06 Quiz

Next, let's try out what you've learned.

Lab intro


Implement a Hub-and-Spoke
Network Using Network
Connectivity Center




Google Cloud

In this lab, you design and implement a classic hub-and-spoke network topology. Your pre-configured environment includes three VPC networks—a central hub and two branches (spoke1 and spoke2). You will create virtual machines (VMs) on each network to test connectivity.

You begin by verifying connectivity between the VMs within and across VPCs. Then, you use NCC to implement a hub and spoke. You retest connectivity to confirm that your hub-and-spoke architecture is fully functional.



Today's agenda



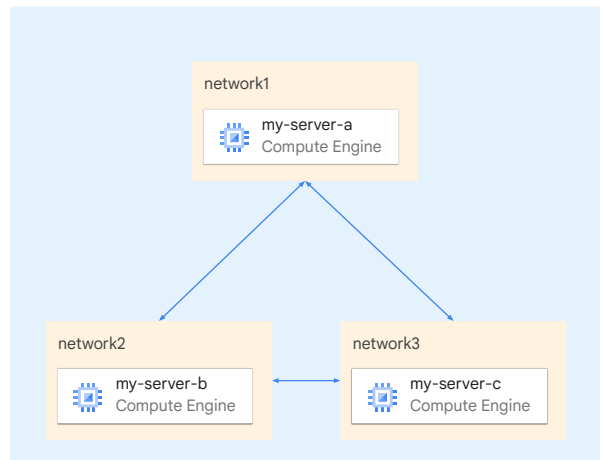
- 01 Hub-and-spoke topology
- 02 Lab: Implement a Hub-and-Spoke Network Using Network Connectivity Center
- 03 [Other topologies](#)
- 04 Getting topology data
- 05 Best practices
- 06 Quiz

Google Cloud

We covered hub and spoke in the previous section. Let's explore other topologies.

Mesh topology

- In a mesh topology, devices or network nodes have multiple interconnected links.
- There are two main types of mesh topology:
 - Full mesh: Every node is connected to every other node.
 - Partial mesh: There is a strategic connection between selected nodes.
- Mesh topologies can work well for applications with many internally connected microservices, such as GKE Enterprise.



Google Cloud

In a mesh topology, devices or network nodes have multiple interconnected links. This contrasts with a hub-and-spoke topology, where devices connect through a central point, that is, the hub.

There are two main types of mesh topology. In a full mesh, every node is connected to every other node. This topology can be useful for GKE Enterprise applications, where the microservices are all internally connected.

In a partial mesh, only select, strategic nodes are connected. For example, for some failover scenarios, a partial mesh is helpful. You create a partial mesh between VPCs in different regions or zones. You can configure routing rules that direct traffic to the active region or zone normally but also configure automatic failover. If a primary region experiences issues, automatic failover to a nearby region or zone using the mesh links keeps your workloads working normally from a client perspective.

Let's consider some of the benefits of a mesh topology.

Mesh topologies provide high availability and resilience. The multiple paths between nodes in a mesh network ensure that if one connection fails, traffic can be rerouted. This makes mesh networks ideal for mission-critical applications that require high uptime.

The Google Cloud infrastructure is designed for scalability. Mesh topologies can easily expand with your cloud environment as you add new nodes or Google Cloud

regions.

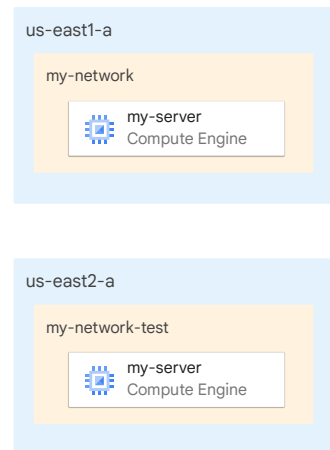
A mesh topology can offer the potential for improved performance. The multiple pathways in a mesh can facilitate better load balancing and reduce network congestion, potentially leading to improved application performance.

The lack of a single central point of failure also aids in security. By distributing traffic across multiple paths, mesh topology can make it harder for attackers to compromise the entire network.

Mirrored topology

A mirrored topology replicates your network environment for different use cases:

- Disaster recovery: provides a failover region to minimize downtime.
- Testing and development: creates isolated environments for experimentation.
- Global workload distribution: distributes traffic across regions for better performance.



Google Cloud

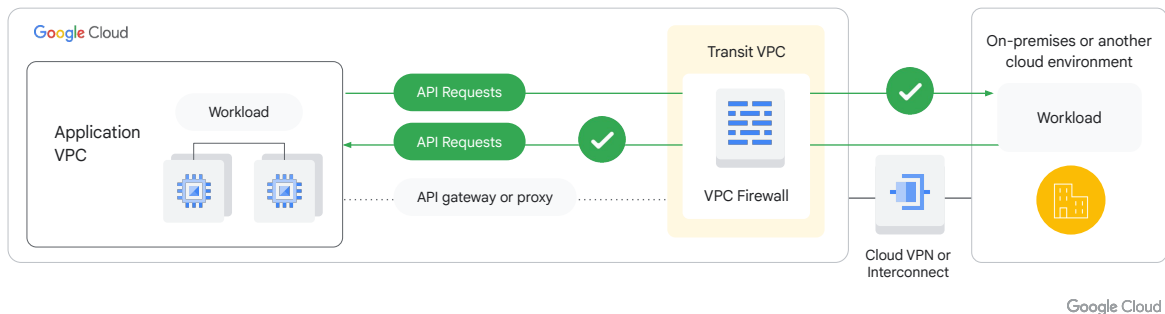
In Google Cloud, mirrored topology means creating a replica of your network infrastructure in another environment or region. Let's consider some use cases. For disaster recovery, an identical standby environment is ready to take over in case of failure. For testing and development, an isolated but otherwise identical configuration as a production environment is available for testing and experimentation. This is shown in the graphics on the slide. Any actions taken in the testing and development environment will have no effect on production. Finally, by distributing traffic across regions, your workloads can take advantage of having an identical environment available nearby, resulting in improved workload performance.

Gating topologies

Gating topologies are great for securing traffic flow due to their fine-grained manner of exposing traffic and services.

There are three types of topologies that restrict access:

- Gated egress: Controls outbound traffic from the cloud.
- Gated ingress: Controls inbound traffic to the cloud.
- Gated ingress and egress: Controls inbound and outbound traffic between hybrid and multi-cloud environments.




Shown here is an example of a type of gating topology. Gating topologies are essential for managing and securing network traffic flows in cloud environments, particularly in hybrid and multi-cloud scenarios. By implementing these gating mechanisms, you gain fine-grained control over data movement between your cloud and external resources, ensuring security and compliance.


There are three types of topologies that restrict access:

- Gated egress: Controls outbound traffic from the cloud.
- Gated ingress: Controls inbound traffic to the cloud.
- Gated ingress and egress: Controls inbound and outbound traffic between hybrid and multi-cloud environments.

More information about each of these topologies can be found in [Gated patterns](#), in the Google Cloud documentation.



Today's agenda



- 01 Hub-and-spoke topology
- 02 Lab: Implement a Hub-and-Spoke Network Using Network Connectivity Center
- 03 Other topologies
- 04 [Getting topology data](#)
- 05 Best practices
- 06 Quiz

Google Cloud

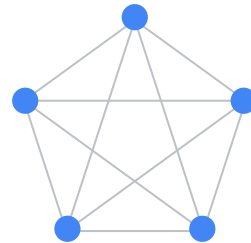
We learned a lot about different topologies. If you are wondering if there was a way to visualize the topology of the network, the answer is yes. Network Topology does exactly that. Let's learn about getting topology data in this section.

Network topology overview

Shows the topology of the network infrastructure.

Presents a graph format where nodes represent the entities and lines represent connections.

Simplifies understanding of complex network relationships and bottlenecks, optimizing traffic flows and troubleshooting network issues.

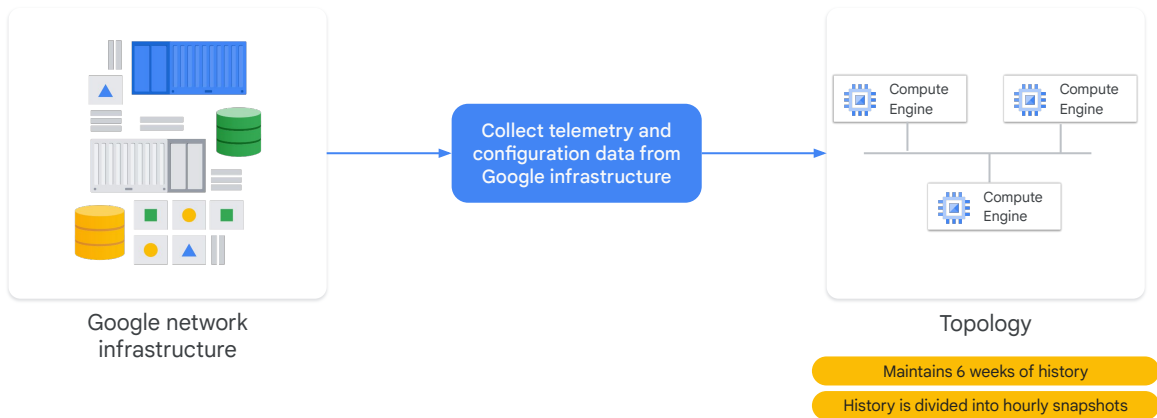


Google Cloud

Network Topology is a visualization tool that shows the topology of the network infrastructure. It serves as a map, illustrating the complex relationships between various components and how they interact.

Network Topology offers an "infrastructure view" showcasing elements like VPC networks, hybrid connectivity, and connections to Google-managed services. It goes beyond static configurations by incorporating real-time operational data, revealing traffic paths, throughput, and the current state of connections between various workloads. By presenting this information in a graph format, where nodes and lines represent entities and connections, Network Topology simplifies the understanding of complex networking relationships. This visual representation aids in identifying bottlenecks, optimizing traffic flows, and troubleshooting network issues more effectively.

How it works



Google Cloud

Network Topology actively gathers real-time telemetry and configuration data from Google's infrastructure to create a visual representation of your network resources. It captures various elements like configuration details, performance metrics, and logs to deduce the relationships between resources within a single project or across multiple projects. By consolidating this information, Network Topology generates a comprehensive graph that accurately depicts your entire network deployment.

Network Topology maintains a historical record spanning six weeks, offering valuable insights into past network configurations and interactions. This history is organized into hourly snapshots, commencing at the beginning of each hour. Each snapshot captures the fundamental entities and their communication patterns within that specific hour. Even if resources, like instances, are created and deleted within an hour, they are still documented in the snapshot for that duration. This ensures a comprehensive historical representation of your network's evolution.

Network topology tools representation

- Entities represent individual resources capable of direct network communication.
- To simplify the visualization of complex networks, base entities are aggregated into hierarchical entities that can be expanded.
- Traffic between entities is represented as lines, connecting entities if at least one side is sending traffic.



Google Cloud

To simplify the view, Network Topology aggregates related resources into hierarchical entities, with each resource type having its own distinct hierarchy.

At the lowest level of each hierarchy are base entities, which represent individual resources capable of direct network communication, such as VM instances or GKE pods. When dealing with complex networks containing numerous base entities, a flat view can be overwhelming. To address this, Network Topology aggregates base entities into higher-level hierarchical entities that can be expanded or collapsed. Initially, the graph displays all base entities aggregated into their top-level hierarchy. For example, VM instances are aggregated into instance groups, then into zones. Meanwhile, GKE pods are aggregated into workloads, then namespaces, and finally into clusters. Each entity, whether base or hierarchical, is represented as a circular node in the graph, and each base entity has its own unique hierarchy. For example, load balancers are organized differently than VM instances within the graph.

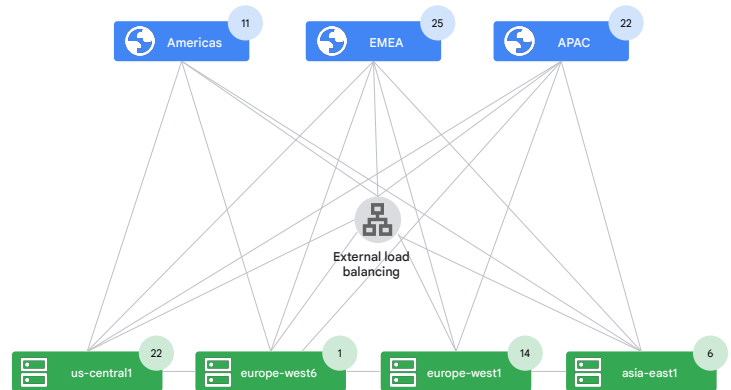
Network Topology visually represents traffic between entities as lines, connecting entities if at least one side is sending traffic. It displays connections across various hierarchy levels, as long as the base entities communicate. For instance, a connection between two regions is shown if at least one VM instance in each region is communicating. The tool supports various traffic protocols (TCP, UDP, ICMP, ICMPV6, ESP, GRE) for specific paths, visualizing traffic within and across VPC networks, between Google Cloud and the internet, and to/from VPN gateways, Interconnect connections, and router appliances. In the GKE view, Network Topology shows traffic within a cluster (between pods on different nodes), between pods on the same node if

intranode visibility is enabled, and between clusters and external IPs.

Use case: Troubleshoot network connectivity



Latency issue causing mobile application to slow down and time out



Google Cloud

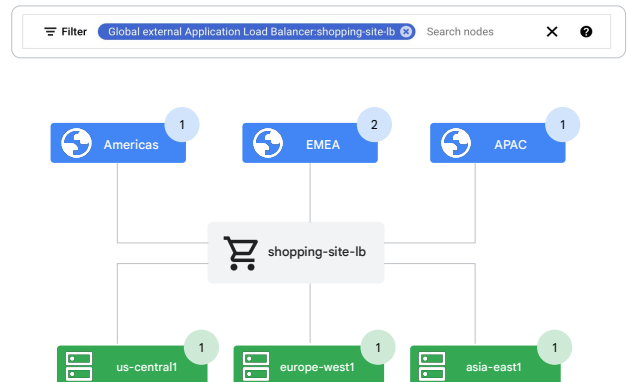
Let's explore a use case. Charlie, a network administrator responsible for a network that includes several load-balanced applications, has been alerted to a latency issue causing the organization's mobile application to intermittently slow down and time out. Multiple users are experiencing this problem, and Charlie has confirmed that no recent application deployments have occurred. The issue is likely due to a change in the network environment rather than a problem with the application itself.

Use the Network Topology tool, which provides comprehensive visibility into the cloud network infrastructure. Describe how its graphical representation of the network topologies enables engineers to quickly understand the connections between resources, identify potential bottlenecks, and troubleshoot network issues efficiently.

Filter to view specific traffic

By filtering the Network Topology view to specifically display traffic for the shopping-site-lb load balancer, you isolate the connections related to the load balancer.

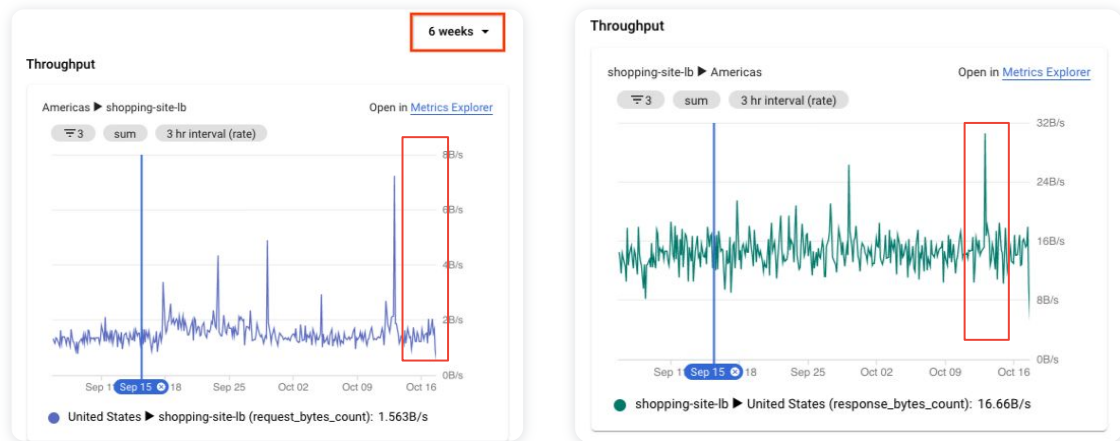
This reveals potential issues in the backend.



Google Cloud

In this scenario, you have a load balancer named shopping-site-lb. Upon noticing increased latency between external clients and the load balancer, you decide to investigate the load balancer's backends. By filtering the Network Topology view to specifically display traffic for the shopping-site-lb load balancer, you isolate the connections related to the load balancer, revealing potential issues in the backend infrastructure that could be contributing to the latency problem.

Review traffic metrics and extend the time series




Google Cloud

Beginning with the external clients in the Americas, you select the traffic metrics between that region and the shopping-site-lb load balancer. Network Topology then displays charts in the details pane, providing information such as ingress and egress traffic between the selected entities, along with key metrics like queries per second (QPS) and HTTP request latency.


Upon reviewing the request latency chart, you notice that the 50th, 95th, and 99th percentile values are all higher than expected. To gain a broader perspective on this latency issue, you extend the time series charts to cover the past 6 weeks for further analysis.

Upon expanding the timeframe, you observe a significant jump in latency that occurred approximately two hours ago, coinciding with the initial reports of the issue. This confirms your suspicion that the increased latency is related to the load balancer.

To investigate further, you navigate to the Load balancing page in the Google Cloud console. Your analysis reveals that one of the instances in the load balancer's backend service was taking longer than usual to respond. By removing this problematic instance from service, you successfully resolve the latency issue and restore normal application performance.



Today's agenda



- 01 Hub-and-spoke topology
- 02 Lab: Implement a Hub-and-Spoke Network Using Network Connectivity Center
- 03 Other topologies
- 04 Getting topology data
- 05 [Best practices](#)
- 06 Quiz

Google Cloud

We learned how to gather topology data. Let's next explore some best practices to consider when exploring the right connectivity options.

Best practices for hybrid cloud environment

- 01 Ensure solution meets the required SLA for performance and uptime
- 02 Scale hub-and-spoke architectures with centralized hybrid connectivity
- 03 Expose applications through APIs using an API gateway or load balancer.
- 04 When using Cloud Load Balancing utilize its application capacity optimization capabilities.
- 05 Use two authoritative DNS systems for private Google Cloud environments.

Google Cloud

Selecting the right network connectivity solution for hybrid or multicloud environments requires careful consideration of several key factors. These include ensuring the solution meets the required service-level agreements (SLAs) for performance and uptime.

For organizations aiming to expand a hub-and-spoke network architecture across multiple VPC networks, establishing centralized hybrid connectivity within a dedicated VPC network, then peering with other projects using custom advertised routes, can be an effective strategy. This approach enables the seamless sharing of both static and dynamically learned routes with peered VPC networks, fostering a centralized configuration that simplifies management and facilitates scalability in your VPC network design.

To enhance application accessibility and management, it's recommended to expose applications through APIs using an API gateway or load balancer. A comprehensive API platform, such as Apigee, can be utilized to simplify this process. This type of platform acts as an intermediary for your backend service APIs, providing features like enhanced security, rate limiting, quotas, and analytics for better control and monitoring of your API usage.

When leveraging Cloud Load Balancing, it's recommended to utilize its application capacity optimization features whenever possible. These capabilities can be valuable in mitigating capacity challenges that often arise in globally distributed applications, ensuring smoother operation and better user experiences.

When deciding where DNS resolution should be performed in a hybrid setup, we recommend using two authoritative DNS systems for your private Google Cloud environment and for your on-premises resources that are hosted by existing DNS servers in your on-premises environment.


Let's ask Gemini ✨

Describe a scenario where a <topology_name> would be the best choice.


What are the advantages and disadvantages of a mesh topology?

Before we wrap up, here is a quick preview of some prompts you can use with Gemini related to network topology.

You can use a prompt such as “Describe a scenario where a <topology name> would be the best choice.” Or “What are the advantages and disadvantages of a mesh topology.”



Today's agenda



- 01 Hub-and-spoke topology
- 02 Lab: Implement a Hub-and-Spoke Network Using Network Connectivity Center
- 03 Other topologies
- 04 Getting topology data
- 05 Best practices
- 06 [Quiz](#)

Google Cloud

Next, let's test your knowledge about this lecture.

Quiz | Question 1

Question

You are designing a Google Cloud network for a large financial services company with strict security requirements. The network needs to isolate sensitive customer data from other resources and limit communication between specific network segments. Which of the following network topologies would be most suitable for this scenario?

- A. Hub-and-spoke
- B. Gated ingress and egress
- C. Mirrored
- D. Mesh

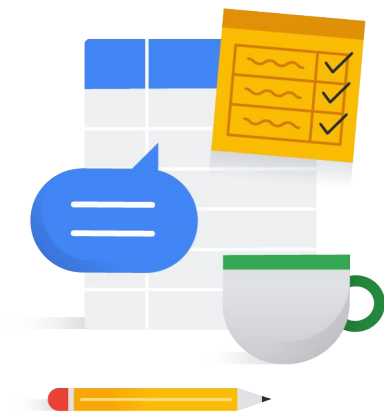
Quiz | Question 2

Question

You are migrating a large ecommerce company's existing on-premises data center to Google Cloud. The on-premises network consists of geographically dispersed regional offices, each with its own network segment requiring secure isolation. However, central management and communication between all regional offices are critical for business operations. Which network topology would *most* effectively address these requirements in Google Cloud?

- A. Hub-and-spoke
- B. Mesh
- C. Mirrored
- D. Gated ingress and egress

Debrief



Google Cloud

This module introduced you to network topologies. We started with the popular hub and spoke model, exploring its structure and applications. Additionally, we covered alternative topologies like mesh, mirrored, and gated. Through a hands-on lab, you explored how to implement a hub and spoke using Network Connectivity Center.