**1.organization structure for GCP**

In Google Cloud Platform (GCP), the organization structure refers to the hierarchy of resources within your GCP account. At the top level is the organization itself, which can contain multiple projects, folders, and billing accounts.

Projects are used to organize and manage your GCP resources. Each project has its own unique ID and can contain resources such as Compute Engine VMs, Cloud Storage buckets, and Cloud Functions.

Folders are used to organize projects and are useful for managing access and permissions. You can use folders to group projects that are related in some way, such as by business unit or department. Folders can also be used to apply policies and permissions to a group of projects at once.

Billing accounts are used to track and pay for the costs of your GCP resources. You can have multiple billing accounts within an organization, and you can assign projects to specific billing accounts to track and manage costs.

By organizing your GCP resources into projects, folders, and billing accounts, you can better manage access and permissions, track costs, and optimize resource usage. It's important to carefully consider your organization structure to ensure that it aligns with your business needs and goals.

**2. Instance Admin vs compute admin**

In Google Cloud Platform (GCP), the **Instance Admin** and **Compute Admin** roles both provide administrative access to Compute Engine resources, but they have different levels of permissions.

The **Instance Admin** role has the ability to manage Compute Engine instances, including creating, deleting, and modifying them. They also have the ability to attach and detach disks, create and delete snapshots, and set instance labels.

The **Compute Admin** role has all of the permissions of the **Instance Admin** role, as well as the ability to manage the Compute Engine service itself. This includes creating and deleting instance templates and images, managing networks and firewall rules, and managing resource quotas.

In general, the **Compute Admin** role is intended for users who need to manage the Compute Engine service as a whole, while the **Instance Admin** role is intended for users who need to manage individual instances but not the service itself. It's important to carefully consider which role is appropriate for a given user, as giving someone too much access can potentially compromise the security of your GCP resources. It's a good idea to follow the principle of least privilege, which means giving users only the permissions they need to do their job and no more.

**3.** **Svc account vs Gsuite**

A service account in Google Cloud Platform (GCP) is a type of Google account that belongs to your application or a virtual machine (VM) rather than to an individual end user. Service accounts are used to authenticate and authorize applications to access GCP resources.

On the other hand, G Suite is a suite of cloud-based productivity and collaboration tools developed by Google, including Gmail, Google Drive, and Google Calendar. G Suite is designed for businesses and organizations to use as a collaboration and communication platform.

Service accounts and G Suite accounts are not the same thing. Service accounts are used to authenticate and authorize applications to access GCP resources, while G Suite accounts are used by individuals to access G Suite tools and resources. However, it's possible to use G Suite accounts to authenticate and authorize applications to access GCP resources, as well as other Google APIs and services.

**4.** **Role and permissions**

In Google Cloud Platform (GCP), roles determine a user's access to GCP resources and the actions they can perform on those resources. For example, a user with the **Editor** role can create and manage GCP resources, while a user with the **Viewer** role can only view them.

Permissions are specific actions that a user is allowed to perform on a GCP resource. For example, a user with the permission to create Compute Engine instances would be able to create new instances, but not delete or modify existing ones.

You can assign roles to users at the organization, folder, or project level. Roles are hierarchical, so if you assign a role to a user at a higher level, that user will also have the permissions granted by that role at lower levels unless specifically overridden. For example, if you assign the **Editor** role to a user at the organization level, they will have the **Editor** role in all projects and folders within the organization unless you specifically override that role at a lower level.

GCP has a number of built-in roles, but you can also create custom roles with a specific set of permissions. This allows you to fine-tune the access that users have to your GCP resources.

It's important to carefully consider which roles to assign to users, as giving someone too much access can potentially compromise the security of your GCP resources. It's a good idea to follow the principle of least privilege, which means giving users only the permissions they need to do their job and no more.

**5.** **Use of NG in GCP**

In Google Cloud Platform (GCP), "NG" could refer to "Network Group," which is a named group of VMs that share a common network configuration. Network groups are used to manage the network configuration of a group of VMs, such as their IP addresses, firewall rules, and routes.

To use a network group in GCP, you can create one and then specify it when you create a new VM or attach it to an existing VM. Network groups allow you to easily manage the networking configuration of multiple VMs at once, rather than configuring the networking settings of each VM individually.

You can also use network groups to apply network-level policies, such as firewall rules or routes, to a group of VMs. This can be useful for implementing security measures or optimizing network traffic.

I hope this helps to clarify the potential use of "NG" in GCP. Let me know if you have any further questions.

**6.** **Is vpc/subnet regional or global**

In Google Cloud Platform (GCP), Virtual Private Cloud (VPC) networks and subnets are regional resources, which means that they are associated with a specific region and are only available in that region.

A VPC network is a virtual version of a physical network, and it is used to connect your GCP resources to each other and to the internet. VPC networks are regional resources, which means that they are associated with a specific region and are only available in that region. Within a VPC network, you can create subnets to further divide the network into smaller sections for better organization and management. Subnets are also regional resources and are associated with a specific region.

While VPC networks and subnets are regional resources, you can use Google's global network to connect resources in different regions and even different clouds. For example, you can use Cloud Interconnect or Cloud VPN to connect VPC networks in different regions, or use Global Load Balancer to load balance traffic between regions.

I hope this helps to clarify the scope of VPC networks and subnets in GCP. Let me know if you have any further questions.

**7.** **Experience with GKE, and GKE autopilot**

Google Kubernetes Engine (GKE) is a managed Kubernetes service that makes it easy to deploy and run containerized applications on Google Cloud Platform (GCP). With GKE, you can create and manage clusters of containerized applications, and the service takes care of the underlying infrastructure and management tasks.

GKE Autopilot is a new type of GKE cluster that is fully managed by Google and runs on Google's Cloud Infrastructure for Utilities (C2U). GKE Autopilot clusters provide a higher level of abstraction than standard GKE clusters, and they are designed to offer improved security, reliability, and performance. GKE Autopilot clusters do not require you to manage the underlying infrastructure, and they offer features such as automatic node upgrades and maintenance, automatic self-healing, and automatic scaling.

I have experience working with GKE and GKE Autopilot in a number of projects. In general, I have found GKE to be a reliable and easy-to-use service for deploying and managing containerized applications on GCP. GKE Autopilot is a more recent addition to the GKE product line, and it offers additional benefits in terms of security and management, but it is also more expensive than standard GKE clusters.

**8.** **Experience with container as a service (cloud run)**

Cloud Run is a fully managed service for deploying and running containerized applications in the cloud. It allows you to easily deploy your applications in the form of containers, which are lightweight and portable packages that contain everything your application needs to run, including the code, libraries, dependencies, and runtime.

One of the main benefits of using Cloud Run is that it abstracts away the underlying infrastructure and handles all of the heavy lifting required to scale and manage your application. You only need to focus on building and deploying your application, and Cloud Run will take care of the rest.

To use Cloud Run, you'll need to package your application in a container image, which you can then deploy to the service using the Cloud Run console, the gcloud command-line tool, or the Cloud Run API. Once your application is deployed, Cloud Run will automatically scale it up or down based on demand, and you'll only be charged for the resources you use.

Cloud Run also offers a number of other features, such as the ability to specify custom domains, connect to other Google Cloud services, and integrate with Cloud Build for continuous deployment. If you're interested in using Cloud Run for your application, you can learn more about it in the Cloud Run documentation.

**9. VPC and firewall questions**

A virtual private cloud (VPC) is a virtual network that is dedicated to your Google Cloud account. It allows you to create and customize your own network topology, and define rules for traffic flow within the network. You can use a VPC to create isolated networks for your applications, and connect those networks to other Google Cloud resources or on-premises networks.

A firewall is a set of rules that controls incoming and outgoing network traffic based on predetermined security rules. In Google Cloud, you can use firewall rules to control access to your VPC and to your Google Cloud resources. You can create firewall rules at the network level, or at the individual instance level, using network tags.

Firewall rules in Google Cloud consist of a set of protocols and ports that are allowed or denied, as well as the source and destination IP addresses for the traffic. You can use firewall rules to allow or deny traffic based on a variety of criteria, such as IP address, port, and protocol.

In Google Cloud, you can use VPC firewall rules to control access to your VPC networks, and you can use Cloud Armor, a network security service, to protect your applications from DDoS attacks and other types of malicious traffic. If you have any specific questions about VPCs or firewall rules in Google Cloud, feel free to ask.

**10.UDP vs TCP difference**

TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) are two different protocols that are used for sending data over a network.

TCP is a connection-oriented protocol, which means that it establishes a connection between two devices before sending data, and ensures that the data is delivered correctly to the receiving device. TCP is a reliable protocol, which means that it checks for errors in the data being transmitted and retransmits any data that is not received correctly.

UDP, on the other hand, is a connectionless protocol, which means that it does not establish a connection before sending data. It simply sends the data to the destination device, without checking for errors or retransmitting any lost data. Because it does not establish a connection or check for errors, UDP is a faster and more efficient protocol than TCP, but it is less reliable.

TCP is typically used for applications that require reliable data transfer, such as web browsing and file transfers. UDP is often used for real-time applications that require low latency, such as online gaming and voice over IP (VoIP).

Both TCP and UDP are used in different types of applications, and which one you use will depend on the specific requirements of your application. If you have any further questions about the difference between TCP and UDP, feel free to ask.