



Cloud Basics

What is the Cloud?

- "The cloud" refers to a network of servers hosted on the Internet to store, manage, and process data.
- These are the tasks traditionally performed by a local computer or server.

Basic Components of Cloud Computing

Servers

- What is a server in the context of cloud computing?
 - A machine, VM, or container instance that hosts services or applications.
 - Example: Amazon EC2, Google Compute Engine, Azure Virtual Machines.

Storage

- Cloud storage is more than just "disk space."
- Types of cloud storage: Object, File, and Block.
 - Object Storage: Ideal for unstructured data.
 - Example: Amazon S3, Google Cloud Storage, Azure Blob Storage.
 - File Storage: Useful for hierarchical data and file systems.
 - Example: Amazon EFS, Google Filestore, Azure File Storage.
 - Block Storage: Suitable for databases and raw disk storage.
 - Block storage is a type of storage commonly used in cloud and enterprise settings, where data is stored in fixed-sized 'blocks.'
 - Each block acts like an individual hard drive and can be controlled as such.
 - Blocks are stored independently and can be configured or accessed in any order.
 - **Examples**
 - Amazon EBS (Elastic Block Store), Google Persistent Disk, Azure Disk Storage.
 - SQL: Relational databases.

- Example: Amazon RDS, Google Cloud SQL, Azure SQL Database.
- NoSQL: Non-relational or distributed database.
 - Example: Amazon DynamoDB, Google Bigtable, Azure Cosmos DB.

Networking

- **What is Cloud Networking?**
 - A specialized area within cloud computing that is concerned with providing connectivity between cloud services, applications, and users.
 - Provides the backbone for data transport within the cloud and to/from external networks.
- **Virtual Private Cloud (VPC)**
 - Essentially a private network within the public cloud.
 - Allows you to have an isolated network space where you can launch cloud resources in a virtual network that you define.
- **Why VPC?**
 - **Isolation and Security:** A VPC isolates your resources from the public internet and other cloud customers. This improves security by reducing unauthorized access.
 - **Network Control:** Customize your own IP address range, create subnets, and configure route tables and network gateways.
 - **Resource Management:** Easier to manage cloud resources that need to communicate with each other within a VPC.
- **Examples of VPC services**
 - **Amazon VPC:** Part of Amazon Web Services (AWS); allows you to launch AWS resources in a custom virtual network.
 - **Google Cloud VPC:** Provides a similar service but within the Google Cloud Platform, focusing on auto mode for automatic routing and custom mode for advanced users.
 - **Azure Virtual Network:** Microsoft Azure's solution; allows for integration with on-premises data centers through VPNs.
- **CDN (Content Delivery Network)**
 - A system of distributed servers designed to deliver content and media to users more efficiently.
 - Helps reduce latency by serving the content from the nearest geographical location to the user.
 - Commonly used to distribute large files, stream video content, and manage high traffic loads.
 - Examples include Amazon CloudFront, Google Cloud CDN, and Cloudflare.
- **Load Balancer**
 - A networking device or software that distributes incoming application or network traffic across multiple servers.
 - Improves application availability and responsiveness by routing requests only to healthy servers.
 - Can distribute traffic based on various algorithms like Round Robin, Least Connections, or custom rules.

- Examples include AWS Elastic Load Balancing, Google Cloud Load Balancing, and Azure Load Balancer.

Both CDNs and load balancers aim to optimize resource use, maximize throughput, and reduce latency, but they operate at different layers and serve slightly different needs.

Software and Middleware

- Implementations of various architectural patterns (we'll talk more about that separately)

Analytics

- Stream processing, data lakes, and analytics engines.
- Example: Amazon Kinesis, Google Dataflow, Azure Stream Analytics.

Monitoring and Security

- Monitoring: Amazon CloudWatch, Google Cloud Monitoring, Azure Monitor.
- Security: Amazon GuardDuty, Google Security Command Center, Azure Security Center.

Containers and Virtualization

Containers: Docker and More

- Containers are an abstraction layer that bundle your code and its dependencies into a single object.
- **Why Containers?**
 - They ensure that an application runs the same, irrespective of where it's executed, making deployments easier and more reliable.
- **Popular Container Solution**
 - Docker is the most commonly used containerization solution.
- **How It Works**
 - Docker containers share the host system's OS kernel but package the application code and dependencies into a single unit.

Virtual Machines (VMs): The Big Picture

- Virtual machines are an abstraction over physical hardware. They virtualize the entire operating system, not just the application.
- VMs are great for running multiple different operating systems on a single physical machine.

- A Hypervisor like VMware or Hyper-V manages multiple virtual machines on the host system.

Quick Comparison: Containers vs VMs

- **Efficiency**
 - Containers are more lightweight because they share the host system's kernel, unlike VMs which have their own OS.
- **Flexibility**
 - VMs are more flexible in terms of the types of applications they can run but at the cost of using more system resources.

By understanding the key differences between containers and virtual machines, you can better decide which is suitable for your specific needs.

Various Ways of Running Applications on Google Cloud

Virtual Machines with Compute Engine

- **What is it?**
 - Virtual machines running on Google's infrastructure.
- **How to Use**
 - Create and customize your VMs via the GCP Console or API.
- **Trade-offs**
 - **Pros:** Full control over the environment, highly customizable.
 - **Cons:** Management overhead, possible resource wastage.

Kubernetes with GKE (Google Kubernetes Engine)

- **What is it?**
 - Managed Kubernetes service for running containerized applications.
- **How to Use**
 - Deploy your containerized applications to a Kubernetes cluster managed by GKE.
- **Trade-offs**
 - **Pros:** Scalability, high availability, strong ecosystem.
 - **Cons:** Complexity, learning curve.

Managed Containers with Cloud Run

- **What is it?**
 - A fully managed platform for running containerized apps.
- **How to Use**
 - Simply upload your container and Cloud Run takes care of the rest.

- **Trade-offs**
 - **Pros:** Easy to use, automatic scaling.
 - **Cons:** Less control compared to GKE or VMs.

Serverless

- **What is it?**
 - A lightweight, event-driven, compute solution for running single-purpose functions.
- **How to Use**
 - Write single-purpose functions that are triggered by various events like HTTP requests or file changes in Cloud Storage.
- **Trade-offs**
 - **Pros:** Easy to deploy, automatic scaling, only pay for the time your code runs.
 - **Cons:** Limited use-cases, less control over the environment.

Summary and Takeaways

- We've covered four primary ways to run your applications on Google Cloud, each with its own trade-offs.
- Your choice will depend on the requirements of your application and the level of control or simplicity you desire.
- By understanding the different methods and their trade-offs, you can make a more informed decision on how best to run your applications in the cloud.