For this assignment, it was too much to digest and reiterate back, in a summary. What I provided was information that I found on each of the cloud provider websites and Linkedin.

|  |  |  |
| --- | --- | --- |
| **Amazon Web Service (AWS) – Orchestrator Services** | | |
| Orchestrator Service | Use Cases |  |
| Amazon Elastic Container Service (ECS) | Run containerized applications or build microservices | Amazon Elastic Container Service (ECS) is a fully managed container orchestration service that helps you to more efficiently deploy, manage, and scale containerized applications. It deeply integrates with the AWS environment to provide an easy-to-use solution for running container workloads in the cloud and on premises with advanced security features using Amazon ECS Anywhere.  <https://aws.amazon.com/ecs/?c=cn&sec=srvm> |
| Amazon Elastic Kubernetes Service (EKS) | Manage containers with Kubernetes | Amazon Elastic Kubernetes Service (Amazon EKS) is a managed Kubernetes service to run Kubernetes in the AWS cloud and on-premises data centers. In the cloud, Amazon EKS automatically manages the availability and scalability of the Kubernetes control plane nodes responsible for scheduling containers, managing application availability, storing cluster data, and other key tasks. With Amazon EKS, you can take advantage of all the performance, scale, reliability, and availability of AWS infrastructure, as well as integrations with AWS networking and security services. On-premises, EKS provides a consistent, fully-supported Kubernetes solution with integrated tooling and simple deployment to AWS Outposts, virtual machines, or bare metal servers.  <https://aws.amazon.com/eks/?c=cn&sec=srvm> |

|  |  |
| --- | --- |
| **Azure – Orchestrator Services** | |
| Layered approach | Azure Container Instances enables a layered approach to orchestration, providing all of the scheduling and management capabilities required to run a single container, while allowing orchestrator platforms to manage multi-container tasks on top of it.  Because the underlying infrastructure for container instances is managed by Azure, an orchestrator platform does not need to concern itself with finding an appropriate host machine on which to run a single container. The elasticity of the cloud ensures that one is always available. Instead, the orchestrator can focus on the tasks that simplify the development of multi-container architectures, including scaling and coordinated upgrades.  <https://learn.microsoft.com/en-us/azure/container-instances/container-instances-orchestrator-relationship> |

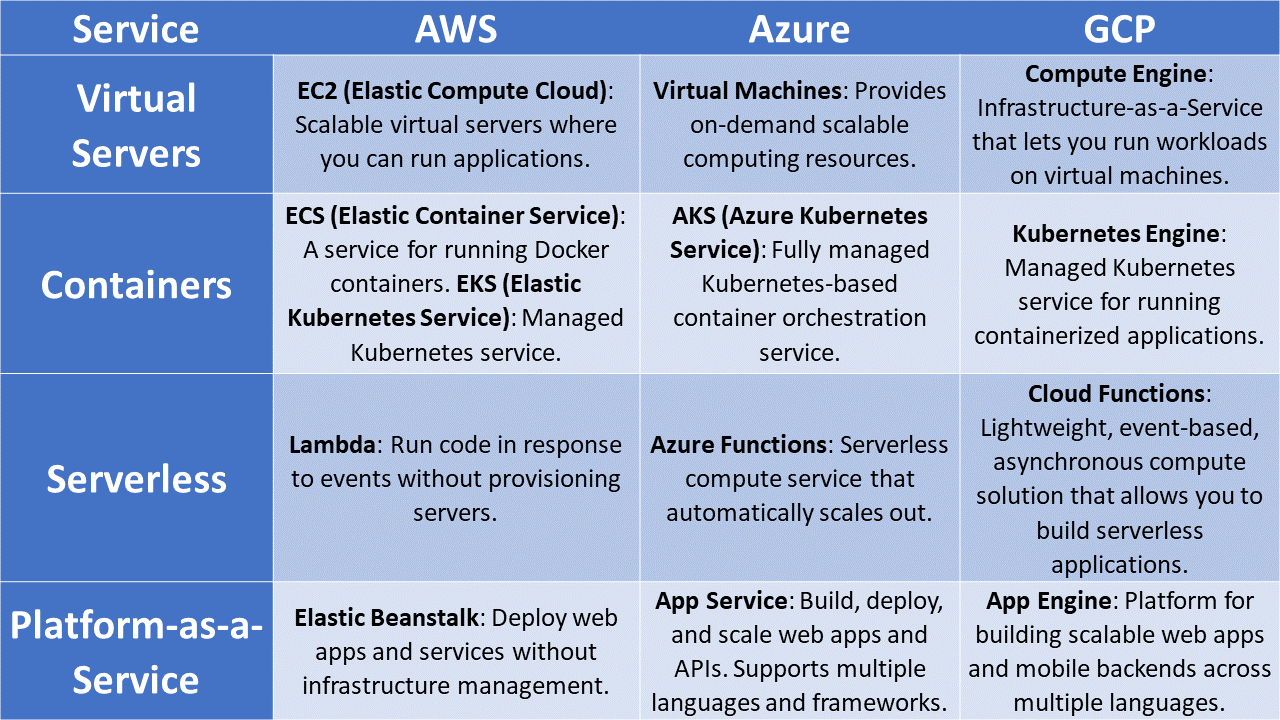
Image Provided by LinkedIn ([https://www.linkedin.com/pulse/comparison-aws-azure-gcp-sanjay-kumar-mba-ms-phd)](https://www.linkedin.com/pulse/comparison-aws-azure-gcp-sanjay-kumar-mba-ms-phd))

Image provided by LinkedIn <https://www.linkedin.com/pulse/comparison-aws-azure-gcp-sanjay-kumar-mba-ms-phd>

