Swathi Anil

8905477

ASSIGNMNET-2

Technical Report

cONTAINERIZATION

SYST8141

1. Briefly explain what GKE is and how it works.
   * What is GKE?
     + **Google Cloud offers a managed Kubernetes solution called Google Kubernetes Engine (GKE).**
     + With Google's infrastructure, you can run and deploy containerized apps at scale with GKE.
     + It's a completely managed environment that makes Kubernetes cluster management easier.
     + The open-source container orchestration platform Kubernetes was first created by Google and automates the deployment, scaling, and maintenance of containerized applications.
   * How GKE works:
     + **Container Runtime Environment**: GKE offers a runtime environment for containers, enabling the apps to operate within them. The configuration, dependencies, and code are all contained within these containers.

The applications operate within containers thanks to the container runtime environment that GKE offers.  Configuration, dependencies, and code are all contained within these containers.

* + - **Orchestration**: The deployment, scalability, and administration of these containers are coordinated by GKE. It makes sure that, for high availability, the apps are dispersed throughout a number of computers, or nodes.
    - **Managed Components:** 
      * **Control Plane**: GKE manages the Kubernetes control plane, which includes components like the API server, etcd datastore, and controller manager. These components handle cluster-wide coordination and management.
      * **Nodes**: GKE provisions and manages the worker nodes in your cluster. Nodes are virtual machines (VMs) that run your containers.
  + **Integration with Google Cloud Services**:
    - Other Google Cloud services, like load balancing, logging, monitoring, and more, are easily integrated with GKE.   
      It offers a security posture dashboard, automatic upgrades, and built-in security safeguards.
    - There are two Models:
      * **Autopilot Mode:** Fully managed, with Google managing the control plane and nodes separately. Only the compute resources that your running Pods request are charged to you.
      * **Standard Mode:** Offers greater flexibility by letting you handle the control plane and nodes independently. All node resources are your responsibility, independent of Pod demands.

1. What are the main advantages of using a managed service such as GKE over on-premises (non-hosted) solutions?
   * The key advantages of using a managed service like **Google Kubernetes Engine (GKE)** over traditional on-premises solutions (Google Kubernetes Engine (GKE) | An Easy Guide, 2024) (Gumaste, 2024):
     + **Simplicity and Abstraction**:
       - **Managed Environment**: The complexity of managing infrastructure is abstracted away by GKE. Networking, server maintenance, and hardware provisioning are not concerns for you.
       - **Focus on Applications**: The staff won't have to worry about doing menial infrastructure duties while using GKE to concentrate on creating and implementing apps.
     + **Scalability and Elasticity**:
       - **Auto Scaling**: GKE automatically scales your application based on demand. It adds or removes nodes as needed, ensuring optimal resource utilization.
       - **Horizontal Scaling**: We can easily scale the applications horizontally by adding more replicas of your containers. This flexibility is crucial for handling varying workloads.
     + **High Availability and Reliability**
       - **Redundancy**: GKE spreads your application across multiple nodes and zones, reducing the risk of downtime due to hardware failures or maintenance.
       - **Automated Failover**: In case of node failures, GKE automatically reschedules Pods to healthy nodes, maintaining service availability.
     + **Security and Compliance**
       - Managed Security upgrades: GKE keeps the cluster safe from vulnerabilities by managing security patches and upgrades for the control plane.
       - Integrated IAM: By integrating GKE with Google Cloud Identity and Access Management (IAM), users and services can have fine-grained access control.
     + Logging and Monitoring:
       - Built-in Monitoring: GKE has built-in monitoring with dashboards for your workloads, nodes, and cluster. Performance, health, and resource utilization can all be monitored.
       - Logging and Tracing: GKE facilitates troubleshooting by integrating with Google Cloud Logging and Cloud Trace.
     + **Cost Efficiency:**
       - **Pay-as-You-Go**: GKE follows a pay-as-you-go model. You pay only for the resources you use, without upfront capital expenses.
       - **Resource Optimization**: GKE optimizes resource allocation, minimizing wastage and maximizing efficiency.
     + **Upgrades and Maintenance**:
       - **Automated Upgrades**: GKE handles Kubernetes version upgrades, ensuring you’re always on a supported and secure release.
       - **Managed Control Plane**: The control plane is maintained by Google, reducing your operational overhead.
     + **Global Reach and Load Balancing**:
       - **Multi-Region Clusters**: By enabling the creation of clusters across several areas, GKE enhances availability worldwide.
       - **Integrated Load Balancing**: Google Cloud Load Balancing and GKE combine to effectively distribute traffic.
2. What are the disadvantages of using such a service?

While **Google Kubernetes Engine (GKE)** offers numerous benefits, it’s essential to consider potential disadvantages as well:

* + - **Cost**:
      * **Pricing:** GKE and other managed services have expenses. The cost of the functionality and convenience may be more than that of maintaining your own on-premises Kubernetes cluster.
      * **Resource Overhead**: GKE abstracts infrastructure management, but you still pay for the underlying VMs (nodes). If your workloads are consistently low, this cost may not be optimal.
    - **Vendor Lock-In**:
      * **Dependency on Google Cloud**: Using GKE ties you to Google Cloud Platform (GCP). Migrating away from GCP can be complex and time-consuming.
      * **Limited Portability**: While Kubernetes is portable, GKE-specific features may not seamlessly transfer to other Kubernetes environments.
    - **Customization Constraints**:
      * **Limited Control**: Certain features are abstracted by managed services, which restricts your ability to modify the node settings or control plane.
      * **Trade-Offs**: You gain simplicity but sacrifice fine-grained control over every detail.
    - **Latency and Data Sovereignty:**
      * **Data Location**: GKE clusters are hosted in specific regions. If your application requires data sovereignty or low-latency access, this could be a limitation.
      * **Network Latency**: Network latency may be introduced during communication between your GKE cluster and other services.
    - **Learning Curve and Expertise:**
      * **Kubernetes Complexity**: While GKE simplifies Kubernetes, understanding Kubernetes concepts is still necessary. Teams need Kubernetes expertise.
      * **Operational Knowledge**: We must become familiar with GKE-specific procedures and best practices.
    - **Service Outages and Downtime:** 
      * **Dependency on Google’s Infrastructure**: GKE’s reliability depends on Google’s infrastructure. Rare outages or maintenance windows can impact your applications.
      * **Mitigation Strategies**:  Implement redundancy and failover mechanisms to minimize downtime.
    - **Security Concerns:**
      * **Shared Responsibility**: We are in charge of protecting your apps and Pods, while GKE takes care of control plane security.
      * **Risk of Misconfigurations**: Incorrect security settings can expose your workloads.
    - **Regulatory Compliance:**
      * **Industry-Specific Regulations**: If the business works in a regulated sector, make sure GKE conforms with all applicable regulations (HIPAA, GDPR, etc.).

1. What does it cost? How does the pricing model work?

Now let's examine Google Kubernetes Engine (GKE) cost information. (Rai, 2023)

* + 1. **Standard Edition:** 
       1. **Pricing**: GKE Standard Edition is priced at **$0.10 per cluster per hour**.
       2. Features: Cost visibility, autoscaling of pods and clusters, fully automated cluster lifecycle management, and automated infrastructure cost optimization are all included.
       3. **Compute Resources**: The price is determined by the machine configuration and is determined by Compute Engine.
    2. **Enterprise Edition :**
* Cost: $0.00822 per vCPU hour is the pricing of GKE Enterprise Edition.
* **Additional Features:**
* Support for several teams and clusters.
* Self-serve activities
* enhanced security
* Management of service mesh configuration
* Observability metrics best practices
* unified experience on the console
* **Hybrid and Multi-Cloud Features**: GKE Enterprise costs are applicable to all managed GKE Enterprise clusters once they are enabled. (GKE Pricing: Schemes and Optimization Strategies, n.d.)
* **vCPU Consideration**: Fees are determined by counting the virtual CPUs in the GKE Enterprise cluster (administrative cluster and control plane nodes not included).
* **Billing**: At the conclusion of every billing cycle, you receive a bill that details charges and use.
  + - **Pay-as-You-Go Pricing**
* Pay-as-you-go pricing is available from GKE Enterprise for both on-premises and public cloud environments.
* US dollars (USD) are used to list prices.
* Here's a summary for several settings.
  + - **Autopilot Mode**:
    - The cost of further cloud installations is determined by the CPU, memory, and temporary storage resources that are deemed essential for the Pods to operate, as determined by the Pod schedule. Each cluster in the autopilot cluster costs $0.10 per hour.

1. Would you recommend using a managed Kubernetes service to companies that are using Kubernetes in production?
   * **Advantages of Managed Kubernetes Services (e.g., GKE)**:
     + **Simplicity and Abstraction**:
       - Managed services abstract away infrastructure management, allowing your team to focus on application development.
       - Simplified cluster setup, upgrades, and scaling.
     + **Scalability and Elasticity**:
       - Auto-scaling and dynamic resource allocation.
       - Easily handle varying workloads.
     + **High Availability and Reliability**:
       - Redundancy across nodes and zones.
       - Automated failover mechanisms.
     + **Security and Compliance:**
       - Managed security updates for control plane.
       - Integrated IAM for access control.
     + **Monitoring and Logging:**
       - Built-in monitoring dashboards.
       - Troubleshooting tools.
     + **Cost Efficiency:**
       - Pay-as-you-go model.
       - Resource optimization.
     + **Upgrades and Maintenance**:
       - Automated Kubernetes version upgrades.
       - Managed control plane.
     + **Global Reach and Load Balancing:**
       - Multi-region clusters.
       - Integrated load balancing.
   * **Disadvantages of Managed Services**:
     + **Cost**:
       - Managed services come with costs.
       - Consider budget constraints.
     + **Vendor Lock-In**:
       - Tied to the specific cloud provider.
       - Limited portability.
     + **Customization Constraints**:
       - Limited control over certain aspects.
       - Trade-offs between personalization and simplicity.
     + **Latency and Data Sovereignty:**
       - Consider data location requirements.
       - Network latency.
     + **Learning Curve and Expertise:**
       - Teams require knowledge of Kubernetes.
       - Recognize GKE-specific functions.
     + Service Outages and Downtime:
       - Dependency on Google’s infrastructure.
       - Implement redundancy.
   * **Recommendation**
     + Managed services like GKE are great options if your company values dependability, scalability, and simplicity.
     + Analyse expenses, knowledge, and particular needs.
     + For flexibility, consider multi-cloud or hybrid systems.
2. Provide a specific company that is using GKE presently (from case studies on the internet e.g., Databricks) and briefly detail why they are using the service and what problems the overcame by using GKE.
   * Here are some specific company that utilize Google Kubernetes Engine(GKE) and the benefits they have gained. (Current: Preparing teenagers for financial responsibility, n.d.)
     + **American Cancer Society (ACS)**:
       - GKE is utilized by the American Cancer Society (ACS) to improve studies on breast cancer.
       - To expedite their study and evaluate tissue photos, they required a platform that was scalable, dependable, and effective. (Shetty, 2022)
     + **Challenges Overcome:**
     + Faster Tissue Image Analysis: ACS used machine learning and computational resources to analyse tissue images 12X faster by utilizing GKE.
     + Effective Resource Use: GKE made it possible for ACS to allocate resources optimally, guaranteeing that machine learning training would be affordable.
     + Enhanced Research Productivity: ACS researchers can concentrate on their work without having to worry about infrastructure administration thanks to GKE. (Google Cloud customers, n.d.)

# References

*Current: Preparing teenagers for financial responsibility*. (n.d.). Retrieved from cloud.google.com: https://cloud.google.com/customers/current/

*GKE Pricing: Schemes and Optimization Strategies*. (n.d.). Retrieved from Cloudbolt: https://www.cloudbolt.io/gcp-cost-optimization/gke-pricing/

*Google Cloud customers*. (n.d.). Retrieved from cloud.google.com: https://cloud.google.com/customers/

*Google Kubernetes Engine (GKE) | An Easy Guide*. (2024). Retrieved from SentinelOne: https://www.sentinelone.com/cybersecurity-101/what-is-google-kubernetes-engine-gke/

Gumaste, P. (2024, March 1). *User’s Guide to Getting Started with Google Kubernetes Engine*. Retrieved from Whizlabs Blog: https://www.whizlabs.com/blog/google-kubernetes-engine-user-guide/

Rai, A. ( 2023, January 25). *Google Kubernetes Engine (GKE) – The Definitive Pricing Guide*. Retrieved from blog.economize.cloud: https://blog.economize.cloud/google-kubernetes-engine-pricing-guide/

Shetty, R. (2022, September 15). *Kubernetes’ Best Practices, Use Cases & Case Studies*. Retrieved from niveussolutions: https://niveussolutions.com/kubernetes-best-practices-use-cases-case-studies/