biqmind

CLOUD MIGRATION USE CASE:

MOVING FROM AWS TO GCP

Facing a highly competitive marketplace and looking at geographical expansion, a multi-media and telecommunications start-up decided to explore moving its customer-facing applications away from Amazon Web Services (AWS) to other cloud platforms. The web and mobile applications through which its customers managed services, paid bills, checked rewards and subscribed to plans were all on AWS. Working on AWS had resulted in a high cost structure, reliance on AWS services and lack of flexibility to benefit from new tools outside the AWS ecosystem.

The development team started to explore options to migrate and standardize on a single cloud provider that could provide better technology and service level agreements (SLAs) across geographical locations cost effectively. They decided on Google Cloud Platform (GCP) for several reasons. Demand-based scaling is extremely easy and responsive, there are options to customize CPU & RAM amounts and; pricing is straightforward and results in lower costs. Being able to leverage Google's network infrastructure capabilities, better hardware and excellent security were other key factors. From an application development perspective, the team had been wanting to try out Elastic Kubernetes Service (EKS) for their containerized applications but did not have the necessary Kubernetes skillsets to support implementation. Google Kubernetes Engine (GKE) provided node health monitoring and auto-repairs by default, enabling automatic updates of Kubernetes Master and Nodes. This would mitigate skill gaps and reduce time required for cluster management and operations.

APPROACH

Plan Upfront and Run a Proof-of-Concept

A project team was assembled to review the risks and benefits:

- How would the new architecture impact and change application management processes?
- What actions were needed to manage costs?
- What were the full set of applications running and what were their dependencies?
- What was the criticality of the applications and data to be moved?
- What privacy and cross-border regulations had to be taken into consideration for data migration?
- What was needed to maintain data integrity and business continuity?
- How could network latencies and operational disruption be kept to a minimum during the migration?

As part of the evaluation, a proof-of-concept (POC) was carried out. The objective was to see if a "lift and shift" approach was possible. The team further wanted to test out running a Kubernetes cluster on GKE. A test development environment was set up on GCP. This allowed the project team to test out their planning assumptions and migration tool set. The POC enabled the team to identify key risks, and fine-tune budget/scope planning for the main project.

Evaluate and Prepare Current State of Infrastructure, Applications and Data

Prior to migration, the team carried out a deep dive and preparation of the current AWS-based architecture:

- Network:
  Routing/IP, subnets, host names and load balancers were mapped out
- Server Setup and Configuration:
  Blue/green deployment, redundancy, size (vCPU, memory), OS and machine types, load balancers in place for auto-scaling were reviewed

- Services:
  AWS services in use were consolidated and mapped against corresponding GCP products and gaps identified
- Data Migration:
  A read replica was created to enable migration from the relational database (MySQL) on AWS cloud to Google Cloud SQL on GCP
- Application Architecture:
  Dependencies were studied and a run-book was created for groups of VMs as certain applications required high reliability and availability

Migrate, Anticipate, Manage

After the initial groundwork, a clear 4 week project goal was defined:

- Move SIT, UAT and Production environments from AWS to GCP
- Create a number of Kubernetes clusters

Main Tasks were:

1. Create GCP Account and ensure team have relevant access to existing AWS instances
2. Configure project and organization on GCP for migration
3. Create relevant roles & service accounts following permission structure on AWS IAM (Identity and Access Management)
4. Setup account in the tool for exporting and migrating VM
5. Install agent tool in AWS EC2 instances to enable live replication
6. Create blueprint (properties of target machine on Google Cloud type of instance, name, subnet etc) for configuration of instance on Google Cloud once the replication is completed
7. Launch machines on GCP in test mode to test continuous data replication between AWS and GCP
8. Complete testing on migrated instance and implement cutover mode to remove all contact with AWS instance

Important Notes:

a. Create servers behind application load balancers in private subnets in different Availability Zones (AZs) to ensure application redundancy across GCP AZs
b. Project requirements may require public facing servers being provisioned with public subnets
c. Put Disaster Recovery plans in place
d. Use Stackdriver for monitoring workloads on GCP
e. Set up elasticity and autoscaling
f. Use templates (configuration as code) on Google's Cloud Deployment Manager to launch and scale new instances
g. Move any unused and static assets to archival storage (cold storage) on GCP (similar to Glacier on AWS)
h. Migrate any data and volumes attached to instances before migrating the instances themselves
i. Use the lift and shift approach for application migration where possible
j. Use VPN if possible as replication using public IP for migration can slow down the migration process

KEY OUTCOMES

Moving an enterprise workload to another cloud provider is not without its challenges, and the team found that besides solving problems, it was important to anticipate and resolve new ones. Unused/redundant instances were identified before migration and deleted to reduce space and cost on GCP. Infrastructure requirements were also re-factored with respect to application dependencies.

The migration was successful with these outcomes achieved:

- Significant cost-savings
- Better and faster infrastructure
- Flexibility to introduce different tools
- Access to a managed Kubernetes platform
- Access to Google's security tools

WHY WORK WITH BIQMIND

Biqmind provides cloud-native specialist resources and software tools that help companies make the migration to cloud possible. Our certified specialists have field experience with large-scale migration projects across a variety of industries. From evaluating existing workloads, developing a strategic plan, performing production deployments, to mentoring your teams, we build a sustainable roadmap and equip your team to effectively adopt and unlock the benefits of new platforms.

biqmind

Contact our enablement team:
GianLuca Carola
gianluca.carola@biqmind.com
Vinod Narayanankutty
vinod.nara@biqmind.com