A Guide to DevOps Implementation on Google Cloud

IT-reliant companies have utilized the DevOps approach swiftly and in iterations to churn out products. While AWS and Azure entered the market sooner, Google entered the fray with its own set of products that support DevOps. The practice, which brings together Development and Operations, demands many resources, which puts a massive onus on the cloud providers.

Additionally, the plethora of tools is bound to befuddle those new to the cloud. While we spoke about DevOps on AWS and Azure, we shall look at Google Cloud and how it supports DevOps practice in this blog.

Overview of Google Cloud

AWS sparked the cloud revolution, and Microsoft took the cue and created Azure. AWS and Azure have become one of their parent companies' most lucrative revenue streams. Google Cloud Platform, the youngest in this race, captured its fair share of the market and clinched the third spot in the cloud race as of now.

GCP’s growth fuelled its compatibility with open-source tools such as Kubernetes and other clouds. While multi-cloud strategies are only spawning, GCP came out with Anthos, which supports hybrid and multi-cloud strategies.

What exactly is DevOps

Due to the flawed working model, development and operations teams were at loggerheads in the olden era. While the development team created a tool, the operations unit would find flaws, sometimes sending the development team back to their drawing board. While perfection was ultimately achieved, the rollout was often delayed. The expenses time was always on the higher side. Finally, after much ado, both the teams were combined.

The production approach changed greatly as the developers and ops members worked in unison to roll out the deliverables as iterative updates. The short and multiple updates allowed the DevOps members to prep the deliverables in a short period with flaws that wouldn’t send them to square one.

However, this approach requires considerable digital resources as collaboration is immensely high. From codes being stored on the cloud, which should allow easy access but secure storage, and to testing environs, the DevOps team is resource-extensive and effort-intensive. The latter has been achieved with GCP’s massive library of digital tools.

It is worth knowing that DevOps has become immensely popular, and it has spawned off various other production approaches such as DevSecOps. So, without further ado, let’s dive into the crux of the blog.

How to proceed with DevOps on GCP?

Google is popular for a great deal of many things. From its search engine to GCP, some know it to be the originator of Kubernetes, the tool most DevOps engineers favor. While AWS and Azure have their own Kubernetes tools, GCP has Google K8S Engine (GKE). The tool is the industry’s first wholly managed Kubernetes tool, allowing its users to realize four different autoscaling K8s. It also lends them multi-cluster support.

What sets it apart from its competition is that one can manage the containers entirely from GKE and rest easy as it is heavily fortified with Identity and Access Management security. In addition, it makes the DevOps life smoother as it comes with the auto-repair option, which automatically fixes a damaged node.

Useful Link: EKS Vs. AKS Vs. GKE: Which is the right Kubernetes platform for you?

As the DevOps team makes progress swiftly, it is imperative to monitor the progress of the stacks, and GCP comes packing with Stackdriver. The tool can be utilized for monitoring purposes, and it provides insights on compute engine infrastructure health. It also provides additional intelligence upon network and storage usages.

The metrics are valuable as the company can view the resources used and the overall health of its infrastructure. It is also widely accepted that GCP's operations are relatively cheaper as Google sweetens the pot with discounts to keep the users hooked to the cloud.

Another critical element in DevOps is Continuous Delivery or Continuous Deployment (CI/CD). While it sounds technical, DevOps breaks down one massive deliverable into many iterative updates. This approach effectively puts the onus on the DevOps team to glacially dole out the updates until the project's completion.

Google's Cloud Build helps in this aspect as one can swiftly develop software in all programming languages and has the option from 15 machine types. Google doubled down on the productivity quotient by ingraining the tool with multi-cloud support.

Imbibing the same spirit is Cloud Deploy which allows DevOps to create a deployment pipeline with GKE at the core. It seamlessly integrates with the DevOps ecosystem and allows you to use Jenkins along with it. While Cloud Build is all about developing, Cloud Deploy is the dedicated CD tool that provides insights into deployment success ratio and frequency. Google layered these tools with adequate security as every tool is fortified with Google Cloud's central security and they can be audited easily using Google's infrastructure.

While all these sound great, users can make a costly effort to create the CI/CD pipeline. Taking cognizance of the expenses, Google itself suggests the usage of Tekton, an open-source platform that helps the DevOps stitch together the CI/CD pipeline. One can build and deploy, after due testing, the deliverables across various cloud platforms and on-premises IT infrastructure.

Useful Link: DevOps Implementation in Manufacturing Sector

While there are various other tools in GCP's arsenal, two other tools of its own are noteworthy. Security is paramount in the age of data breaches, and Google has fired on all cylinders by bringing out two products: Artifact Registry and Binary Authorization. The former is a container registry from which one can manage and secure the infrastructure artifacts.

In addition, it is a hub from which one can manage the containers and use Google's tools and runtime applications. The Registry also supports native artifact protocols, and this support makes it relatively easy to use with the CI/CD pipeline.

Binary Authorization enables the deployment of workloads. It is effectively a security mechanism that ensures only trusted workloads are deployed. The mechanism can enforce signature validation, and this enforcement shall lend the DevOps team and organization itself better control over who alters or modifies the existing infrastructure. The verification will remove the chances of unauthorized changes, which can damage or render the infrastructure useless.

In conclusion

AWS, Azure, and GCP are in a league of their own. Each of them has exceptional tools which propel productivity. However, restricting oneself to one cloud will only be counter-productive, and one should consider adopting a multi-cloud strategy to keep up with the competition.

While multi-cloud sounds enticing, it brings on the potential risk of draining away the monetary assets if you rope in unwanted tools. This aspect should motivate you to reach out to Veritis, one of the leading figures in **IT consultancy**. Veritis has advised creating unique, cost-effective solutions without compromising anywhere. So, reach out to us and get your solution to take you to new heights.

Services