SERVERLESS COMPUTING
IS YOUR
CLOUD INFRASTRUCTURE
FUTURE-READY?

Serverless computing, a witty misnomer though, has gained tremendous traction in the past few years. No wonder, the streaming shows that have captured your imagination are powered by this cloud computing model. Yes, the world's leading streaming entertainment service company, Netflix is one of the pioneers to adopt serverless architecture built on AWS Lambda.

In this blog we are going to discuss the major players in this realm, why the world is moving towards serverless, some of the challenges that hinders its adoption and more And we'll also talk about the fabulous four with whom you can team up and get started with your serverless goals. So, let's begin!

THE KEY PLAYERS IN SERVERLESS

When it comes to serverless computing, there are three major players in the market: AWS, Azure and Google Cloud. As technology innovation leaders have already forecasted that "the future is serverless", other big players including Oracle, IBM, SAP and Alibaba have also joined the bandwagon.

According to Datadog, organizations that have invested in serverless technologies are using at least one of the following:

- ◆ AWS: AWS Lambda, AWS App Runner, ECS Fargate, EKS Fargate
- ◆ Azure: Azure Functions, AKS running on Azure Container Instances
- ◆ Google Cloud: Google Cloud Functions, Google App Engine, Google Cloud Run

AWS Lambda is the oldest among these services and is the most popular among developers. It offers a wide range of features and pricing plans to suit any budget. All these services offer robust security and scalability, making them ideal for serverless application development.

WHY ARE ENTERPRISES TURNING TO SERVERLESS ARCHITECTURE?

Let's find out from the IT executives, development executives and developers who have had real-world experiences after implementing a service development approach that included serverless.

In 2021, the IBM Market Development & Insights (MD&I) team conducted a series of surveys on techies of large and mid-market companies. The respondents shared their real-world experiences they have had by going serverless.

BENEFITS FOR APPLICATION DEVELOPMENT

◆ Reduced costs related to managing or running servers, databases or app logic
◆ Improved application quality or performance
◆ Greater flexibility to scale resources up or down automatically
◆ Faster application deployment or rollout of new features

OVERALL BUSINESS BENEFITS

◆ Reduced overall development costs
◆ Better security of company and customer data
◆ Faster time to market or response to changes in the marketplace
◆ Better security of company or customer data

With benefits, also come challenges; 53% of respondents have cited "expertise in serverless architecture is expensive or difficult to find". If you find yourself in that bracket then please get in touch with our Cloud computing experts to discuss your business needs.

The key findings of a research published by Gartner Inc. in April 2022 revealing the key drivers for serverless architecture adoption and commonly cited limitations.

SERVERLESS CLOUD COMPUTING	
KEY DRIVERS FOR ADOPTION	**COMMONLY CITED LIMITATIONS**
✅ Operational simplicity	✅ Vendor lock-in
✅ Platform elasticity	✅ Low degree of operational control
✅ Cost savings	

Source: Gartner 2022

CHALLENGES TO ADOPTION OF SERVERLESS ARCHITECTURES

So we know about the benefits that the serverless architecture brings to your business, however, it brings along some challenges as well. The serverless computing experts at IBM have categorized these challenges into four areas and summarized in the table below:

SERVERLESS COMPUTING CHALLENGES CATEGORISED INTO FOUR KEY AREAS		
AREA	CHALLENGES	MEETING THE CHALLENGE
OPTIMIZATIONS	Reduce overhead of allocating and releasing compute resources	Requires design improvements of cloud system components (Kubernetes enhancements)
	Reduce service latencies	Requires design improvements of cloud system components (Kubernetes enhancements)
	Enable suspending worker nodes	Requires defragmentation of used resources (to be supported by Knative, Kubernetes)

DESIGNING FOR SERVERLESS	Externalizing service flow events	Requires event-driven design of services
	Dealing with serverless service latencies	Requires latency-aware service designs
	Event storms and the possibility of events dropping or becoming irrelevant	Requires better support from cloud systems (Knative) and service design awareness
FUNCTIONS AND WORKFLOWS	Standardization of serverless	Requires serverless communities to meet industry standards
	A common representation of flows between serverless services	Might require a complete workflow language
SECURITY	More control moves from users to cloud automation	Clouds need to take a more active role in securing services that might be vulnerable
	Workload subdivides to more granular services, resulting in a greater attack service	Add automation and revision control to reduce configuration drift and support frequent service updates
	Network boundaries fade away	Add security-behavior monitoring and controls (Knative, Kubernetes)

Now, to explore this futuristic technology, you can experiment with some of the open source projects available. Here are four open source serverless computing platforms to get started with serverless computing.

TEAM UP WITH THE FABULOUS FOUR

Let's take a look at four popular, feature-rich, Kubernetes-based open source serverless computing platforms that are worth exploring to kickstart your journey towards adopting serverless architecture.

- ◆ Nuclio Serverless Platform
- ◆ OpenFaaS Serverless Platform
- ◆ Knative Serverless Platform
- ◆ Kubeless Serverless Platform

NUCLIO SERVERLESS PLATFORM

Nuclio is an open-source project that enables developers to easily create, deploy & manage serverless functions. It provides a streamlined experience for building and deploying data science pipelines as well as other serverless applications.
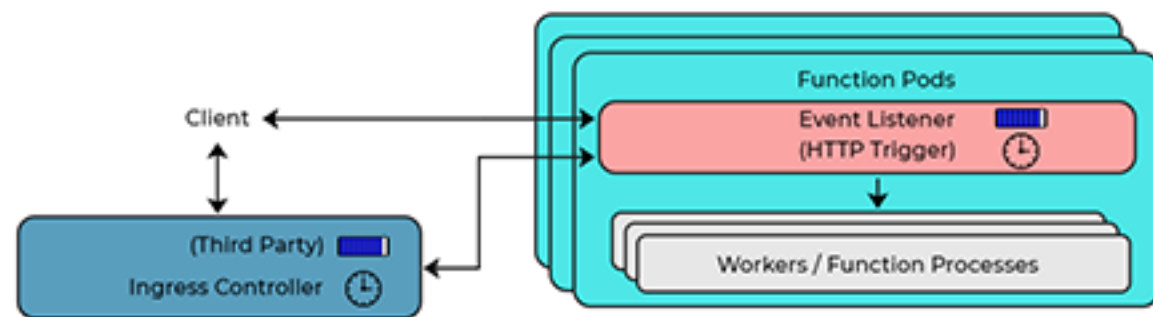
Working model for Nuclio serverless platform

By leveraging the power of Kubernetes, Nuclio allows users to quickly spin up, scale down, and terminate their data science applications on demand. This makes it ideal for use cases that require rapid scaling and high availability, such as machine learning models or real-time analytics.

Nuclio also provides an intuitive web-based interface for monitoring, debugging and optimizing serverless functions, making it easy for data scientists to quickly identify and fix bottlenecks. With Nuclio, data scientists can automate the entire data science pipeline from data ingestion to model deployment with ease.

SALIENT CHARACTERISTICS OF NUCLIO

Let's have a look at the key components of Nuclio as depicted in the image: This platform allows invocation of a 'function' pod directly from an external client without requiring any ingress controller or API gateway. The function pod comprises of two types of processes:

1. Event-listener – In this type of process, you can control the way long events are queued by configuring a timeout parameter
2. One or more worker (user-deployed function) processes

You can set up the number of worker processes as a static configuration parameter which in turn allows:

1. The function pod to run a desired amount of function instances as different processes
2. Parallel execution on a multi-core node.

The unique feature of this framework is its 'Processor' architecture that offers work parallelism. This means that multiple worker processes can be run concurrently in each container.

OPENFAAS SERVERLESS PLATFORM

OpenFaaS is an open source project that provides a serverless platform for developing & deploying cloud applications. It enables you to quickly create and deploy functions, microservices, and other applications in a matter of minutes.
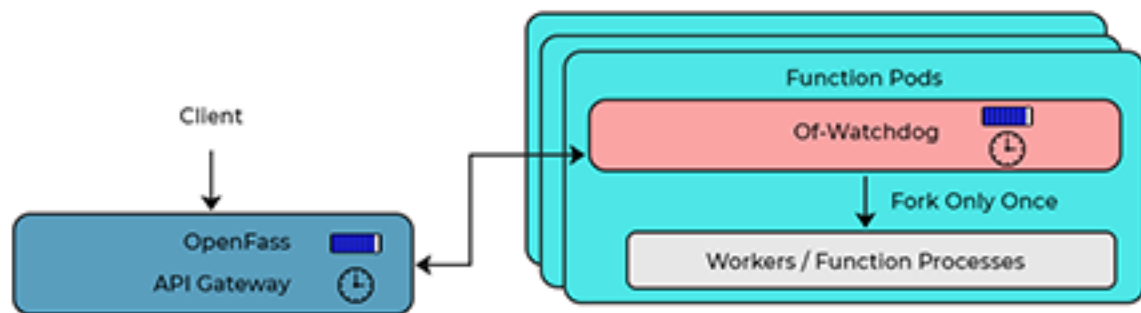
Working model for OpenFaas serverless platform

This serverless computing platform is built on top of Docker and Kubernetes & is designed to be lightweight, easy to use, & highly scalable. With OpenFaaS, developers can create applications that are able to run on any cloud provider and can be quickly updated with new features.

OpenFaaS is also designed to be secure and resilient, making it a great choice for applications that need to remain available even when under heavy load. OpenFaaS is a great tool for developers who want to quickly and easily build cloud applications that can scale with demand.

SALIENT CHARACTERISTICS OF OPENFAAS

The API gateway which serves the following three primary functions:

1. It provides external routing (i.e. the Kubernetes cluster can access the functions from outside)
2. It gathers metrics and interacts with the Kubernetes orchestration to provide scalability
3. It can be scaled to multiple instances or can be replaced by a third-party ingress controller

The function pod houses the following two processes:

1. 'Of-watchdog' – a tiny Golang HTTP server
2. User deployed function process

Of-watchdog serves as an entry-point to receive HTTP requests and forward them to the function process. It can be operated in 3 modes based on use cases:

1. HTTP mode
2. Streaming mode
3. Serializing mode

In the first case, that is the HTTP mode, the single instance (worker) forked in the beginning of the function pod is kept warm throughout the entire lifecycle of the function.

However, in both the streaming mode and serializing mode, a new function instance (worker) is forked for every request. This in turn results in a significant cold-start latency that impacts the throughput.

KNATIVE SERVERLESS PLATFORM

Knative is an enterprise-grade platform for building and running serverless, cloud-native applications. It enables developers to quickly create, deploy, and scale applications on any cloud or Kubernetes-based environment.
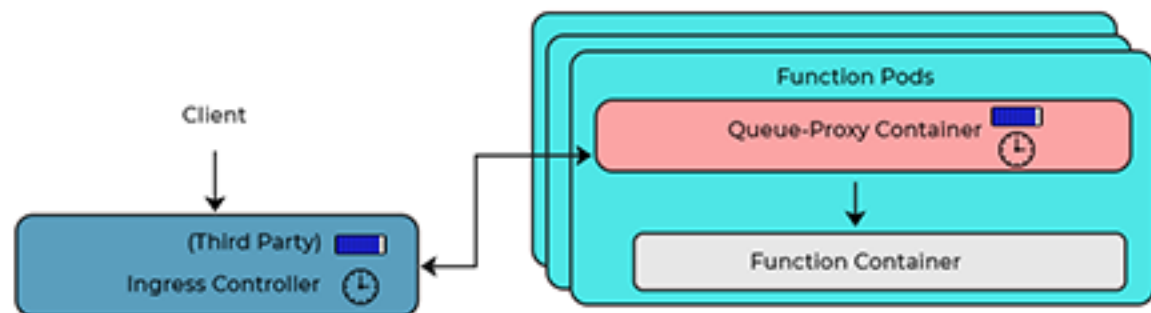
Working model for Knative serverless platform

With Knative, developers can focus on building applications and leave the rest to the platform. Knative handles the operational and infrastructure needs of applications, such as scaling, routing, logging, and monitoring. This allows applications to be more flexible and responsive to changes in demand while also reducing the cost of running them.

This open source serverless computing is quickly becoming a popular platform for cloud-native application development, as it simplifies many of the complexities associated with developing and running applications in the cloud.

SALIENT CHARACTERISTICS OF KNATIVE

The Function pod is a primary component of Knative. Each function pod comprises two containers – i) the queue-proxy container and ii) the function container.

The queue-proxy container – It queues incoming requests and forwards them to the function container. It handles traffic bursts and timeout of queued requests. It also collects metrics and stores them in the internal HTTP server.

One of the distinguishing features of Knative is the 'panic mode' scaling mechanism showcased by the autoscaler component. In this mode the autoscaler quickly scales the function instances (up to the maximum configured limit or ten times the current pod count) to become more responsive to the sudden spikes of traffic (two times the configured threshold value or the desired average traffic).

Another distinct feature of Knative is scale to zero which automatically turns off all inactive functions in containers, and turns back on when there is a demand for them. Thus, scale to zero reduces resource usage as well as costs over time.

KUBELESS SERVERLESS PLATFORM

Kubeless is another event-driven serverless computing platform built on Kubernetes. It provides developers with the ability to quickly and easily deploy functions as needed, without having to worry about setting up and managing servers or other infrastructure.
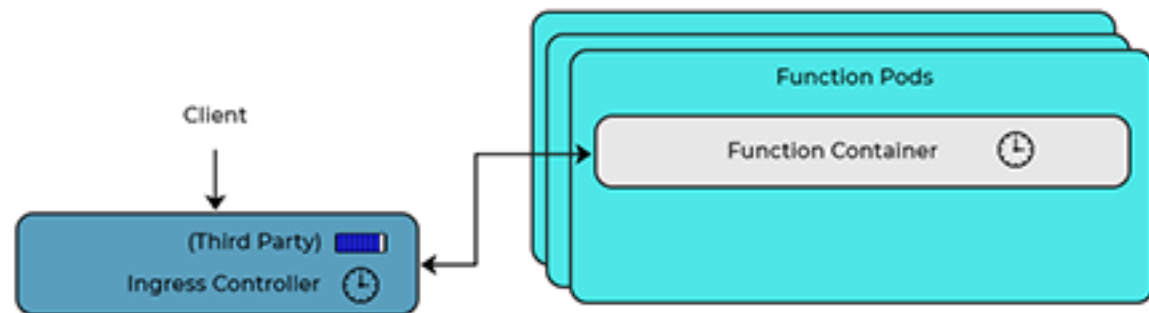
Working model for Kubeless serverless platform

This open source framework also supports a range of programming languages, including Python, Node.js, Ruby and Java. It allows developers to build and deploy functions with ease, while taking advantage of the scalability and reliability of the Kubernetes platform.

It is a great option for developers who are looking for a way to quickly and easily deploy their applications without having to worry about managing servers or dealing with complex infrastructures.

SALIENT CHARACTERISTICS OF KUBELESS

One of the unique features of Kubeless is the ability to create a Custom Reference Definition, or Function as a common resource. In the background, a controller forms and watches over these custom resources and launches them on-demand.

Kubeless allows you to deploy functions without building containers. You can call these functions either via regular HTTP(s) calls or it can be triggered by events submitted to message brokers.

There are three Kubeless functions available as of now:

1. HTTP triggered (an HTTP endpoint will be exposed by the function)
2. Pubsub triggered (managed with Kafka cluster – a built-in package that comes with the installation package of this framework)
3. Schedule triggered (function will be called on a cron job)

Unlike Knative, Kubeless does not support scale to zero but leverages Kubernetes HPA for auto-scaling.

Open source serverless platforms allow you to mix & match different open source services while managing & deploying self-hosted clouds. However, they do come with some challenges such as:

- ◆ Learn and acquire setup expertise needed for integration of different services
- ◆ Management and maintenance of the service infrastructure which cloud service providers take care in proprietary platforms
- ◆ Lack of technical expertise

However, these Kubernetes-based open source platforms have rich and supportive communities you can always reach out to anytime you feel stuck.

THE FUTURE WITH SERVERS: LESS IS MORE ...

To conclude, serverless computing is an extremely powerful technology that offers a wide range of benefits, including scalability, cost savings, and speed of development. If you are looking to get started, many cloud service providers offer managed serverless services that can help simplify the process.

Alternatively, there are open source platforms like Nuclio, OpenFaaS, Knative or Kubeless that allow users to build and deploy their own serverless applications. No matter which platform you choose, serverless is here to stay, and it looks like it will be the dominant model for cloud computing in the future. The efficiency of serverless computing can also contribute towards your sustainability goals. To go serverless and embrace green cloud computing, visit our page and get in touch with our cloud computing experts.

If you haven't leveraged serverless computing yet, it's high time! You are missing the opportunity to enjoy increased productivity and cost savings while being on cloud (nine).