Available Online at http://www.recentscientific.com

CODEN: IJRSFP (USA)

International Journal of Recent Scientific Research
Vol. 9, Issue, 8(B), pp. 28374-28375, August, 2018

ISSN: 0976-3031

International Journal of Recent Scientific Research

DOI: 10.24327/IJRSR

Research Article

SERVERLESS COMPUTING

Matthew N. O. Sadiku*, Shumon Alam and Sarhan M. Musa

Electrical & Computer Engineering Department, Roy G. Perry College of Engineering Prairie View A&M University Prairie View, TX 77446

DOI: http://dx.doi.org/10.24327/ijrsr.2018.0908.2449

ARTICLE INFO

Article History:

Received 13th May, 2018
Received in revised form 11th June, 2018
Accepted 8th July, 2018
Published online 28th August, 2018

Key Words:

Serverless computing, serverless computation, serverless architecture, function as a service

ABSTRACT

In spite of its name, serverless computing does not actually involve running code without servers. Serverless computing allows one to build and run applications without thinking about servers. It has emerged as an approach for hosting applications in the cloud. It has been regarded as a very efficient way to use cloud resources. This paper provides a brief introduction to serverless computing.

INTRODUCTION

Traditionally, applications were written and run on servers which ran all the time. Soon the servers could not handle the enormous requests. Most of the servers are not utilized to their maximum capacity. Consequently, power is wasted while the servers remain idle. Today, all kinds of businesses are using the cloud without giving thought to the underlying infrastructure issues. This new consumption model has led to serverless computing. Serverless computing (SC) has emerged as a new paradigm for the deployment of applications and services. It is also known as Function as a Service (FaaS). It is a new form of cloud computing which is widely accepted as the dominant paradigm in computing. In SC, services are developed as functional service units. Amazon Web Services first announced their Lamda platform in 2014. Lambda ushered in the era of serverless computing. All other big cloud providers (such as including Microsoft, Google, and IBM) have rushed their own versions of Lambda to market. The SC has been quickly adopted by cloud providers because it relieves them from the need for provisioning, scaling, and operating the underlying infrastructure [1].

Features

The term "Serverless" is confusing and does not mean "no servers." For the consumer of the service, there are no virtual machines exposed. In serverless applications,server processes are running somewhere, but the difference compared to normal approaches is that the organization building and supporting a '"serverless" application is not looking after that hardware or those processes [2]. Serverless computing does involve servers, only the implementation and management is different. The developers need not worry about servers. Although your applications run on servers, you need not own, rent, manage or maintain the servers.

SC can be considered as pay-as-you-go computing as you only pay for the time your code is running. Serverless may be regarded as the union of two ideas: Backend as a Service (BaaS) and Functions as a Service (FaaS). It may also be viewed as an evolution of cloud computing service models – from infrastructure-as-a-service (IaaS) to platform-as-a-service (PaaS) to function–as-a-service (FaaS).

Corresponding author:* **Matthew N. O. Sadiku

Electrical & Computer Engineering Department, Roy G. Perry College of Engineering Prairie View A&M University Prairie View, TX 77446

SC is characterized by the following features [3]

- Built using services that support elasticity and fault tolerance
- Automatic scaling, scales up and down instantly
- Integrated security
- Charges for actual usage
- Pay-as-you-go fee structure
- Developer-productivity centric
- Reduced time to market

The interaction between client and serverless architecture is shown in Figure 1, where the black box may contain one or more servers [4].

Benefits

Implementing serverless architecture is economically viable and technically easier. It boosts developer productivity. Other benefits of serverless computing include [5]

- No server management: In serverless computing, you do not need to provision and manage any servers.
- Reducing costs: Everything in serverless architecture is completely managed by the provider itself, eliminating the needs for system administration. Billing is proportional to usage, not reserved capacity.
- Near-Limitless Compute: The consumers have enormous compute power available at their fingertips.
- Simple deployment: Deployment is very different from traditional systems since we have no server applications to run yourselves. By using a serverless architecture, you can transform ideas to reality in few hours.
- Built-in-scaling: Serverless offerings have built-in scalability just like cloud services. Once you pay for the service usage, the serverless architecture will grow or shrink based on demand.

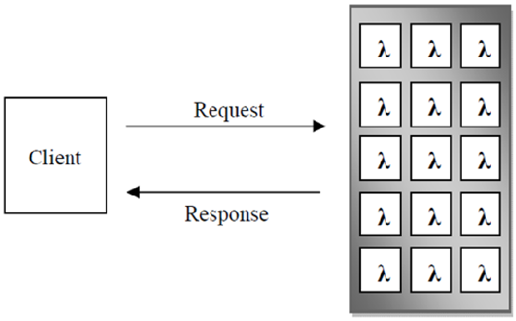

Figure 1 The interaction between client and Lambda serverless architecture [4].

Challenges

As an emerging approach, there are many challenges to be dealt with when developing serverless solutions. Serverless components may not be available in some places. Other challenges include [6].

- Architectural complexity: It gets cumbersome to manage too many functions.
- Problems due to third-party API system: Giving up system control while implementing APIs can lead to system downtime, forced API upgrades, loss of functionality, unexpected limits and cost changes.
- Lack of operational tools: The developers are dependent on vendors for debugging and monitoring tools. The basic debugging operations are not available with serverless functions.
- Lack of support for widely used languages: This challenge will gradually disappear with time because serverless technologies are maturing with support for a wider number of languages.
- Vendor lock-in: Working in a serverless environment can make application to be highly dependent on the platform.

CONCLUSION

Serverless computing is sometimes referred to as an event-driven cloud execution model. It will make sense for some companies, and not with others. Once people understand the financial benefits of using serverless computing, there will be growth in implementation and adoption.

The application of serverless computing is an active area of research and development. SC has proved to be a good fit for IoT applications [7]. Soon we can expect to see companies start to experiment with SC. Serverless computing is a game-changer and is here to stay.

References

1. A. Glikson, S. Nastic, and S. Dustdar, “Deviceless edge computing: Extending serverless computing to the edge of the network,” Proceedings of the 10th ACM International Systems and Storage Conference, Haifa, Israel, May 2017.
2. “Serverless Architectures,” https://martinfowler.com/articles/serverless.html
3. “Serverless computing: A compelling opportunity for today’sdigitalenterprise,”https://www.tcs.com/content/dam/tcs/pdf/technologies/cloud/abstract/Serverless%20Computing.pdf
4. S. Dash and D. K. Dash, “Serverless cloud computing framework for smart grid architecture,” Proceedings of the IEEE 7th Power India International Conference, November 2016 .
5. “Serverless computing, explained,”https://www.booleanworld.com/serverless-computing-explained/
6. “What is serverless architecture? What are its criticisms and drawbacks?”https://medium.com/@MarutiTech/what-is-serverless-architecture-what-are-its-criticisms-and-drawbacks-928659f9899a
7. G. McGrath and P. R. Brenner, “Serverless computing: Design, implementation, and performance,” Proceedings of the IEEE 37th International Conference on Distributed Computing Systems Workshops, 2017, pp. 405-410.

****