Original Article

An Efficient Approach of Security Risk Mitigation in Cloud

Dr.D. Ponmary Pushpa Latha[1], D. Joseph Pushparaj[2], Dr.S. Thanga Helina[3]

[1]Associate Professor, Department of Digital Sciences, Karunya Institute of Technology and Sciences, Coimbatore. E-mail: ponmarymca@gmail.com

[2]Assistant Professor, PSN College of Engineering and Technology, Tirunelveli. E-mail: er.joseph.raj@gmail.com

[3]Assistant Professor, Department of Commerce and International Trade, Karunya Institute of Technology and Sciences, Coimbatore. E-mail: thangahelina@karunya.edu

Abstract

Cloud computing is the emerging fields. It has been emerged with new dimensions because of the various services and computing reference model. In serverless architecture, the server is executed by the cloud provider and also manages the resource allocation dynamically. Many organization has adopted the serverless architecture for their work. Product functionality has been concentrated more on serverless architecture. It is a new era of cloud computing technologies and also gives the maturity in the cloud computing. Organization does not have the job of security patches due to the serverless computing in the OS and application servers. Scalability, flexibility, and reduced cost are the benefits of serverless computing. Purchasing, provisioning, and managing backend servers are the worries are not needed by the serverless architectures. The scope of this research is to analyse security flaws in traditional architecture and mitigate those flaws by using serverless architecture, and to implement security in it.

Keywords: Cloud computing, Security.

DOI: 10.47750/pnr.2022.13.S03.055

INTRODUCTION

Cloud computing is the use of off-site systems and it has the following things: store, manage, process and communicate information. Based on the cloud computing services provided, examples of cloud computing is defined[3-4]**.** SaaS or Software as a Service is a cloud service. Google Apps is an example for this. IaaS or Infrastructure as a Service is an another service. Dropbox is an example for this. PaaS or Platform as a Service is another service. Google App Engine and Heroku is an example for this. Serverless Computing uses the server on the cloud. Elasticity, easier maintenance and provided by them. The servers are vitualized. Instead of giving on-site server. They provide multiple servers in multiple locations and they are connected with each other. Data Storage. Backup & Disaster Recovery, Increasing Bandwidth etc., are the additional examples of cloud computing.

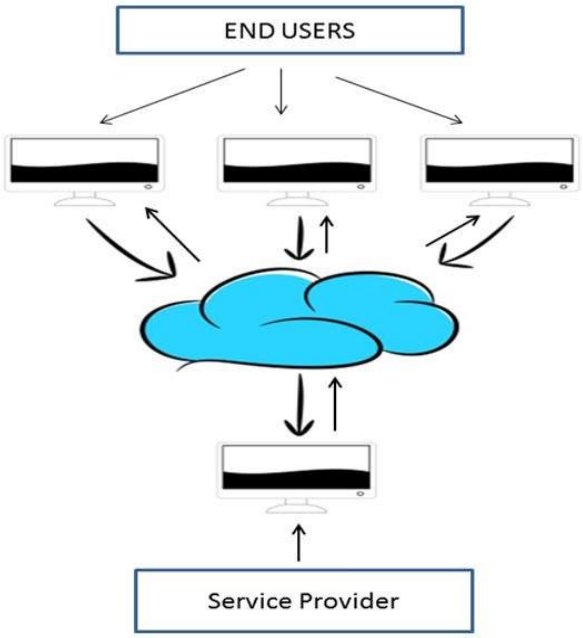

Fig. 1. Cloud Computing

Amazon Web Service (AWS)

Amazon Web Service (AWS) is a platform for cloud and it gives more reliable, scalable and cost-effective cloud platform. It provides so many products such as compute, delveloper tools, management tools and IoT etc., There are n number of services are available in AWS and one of the service is AWS LAMBDA. Self- contained applications and functions are created by this AWS LAMBDA[1-2].

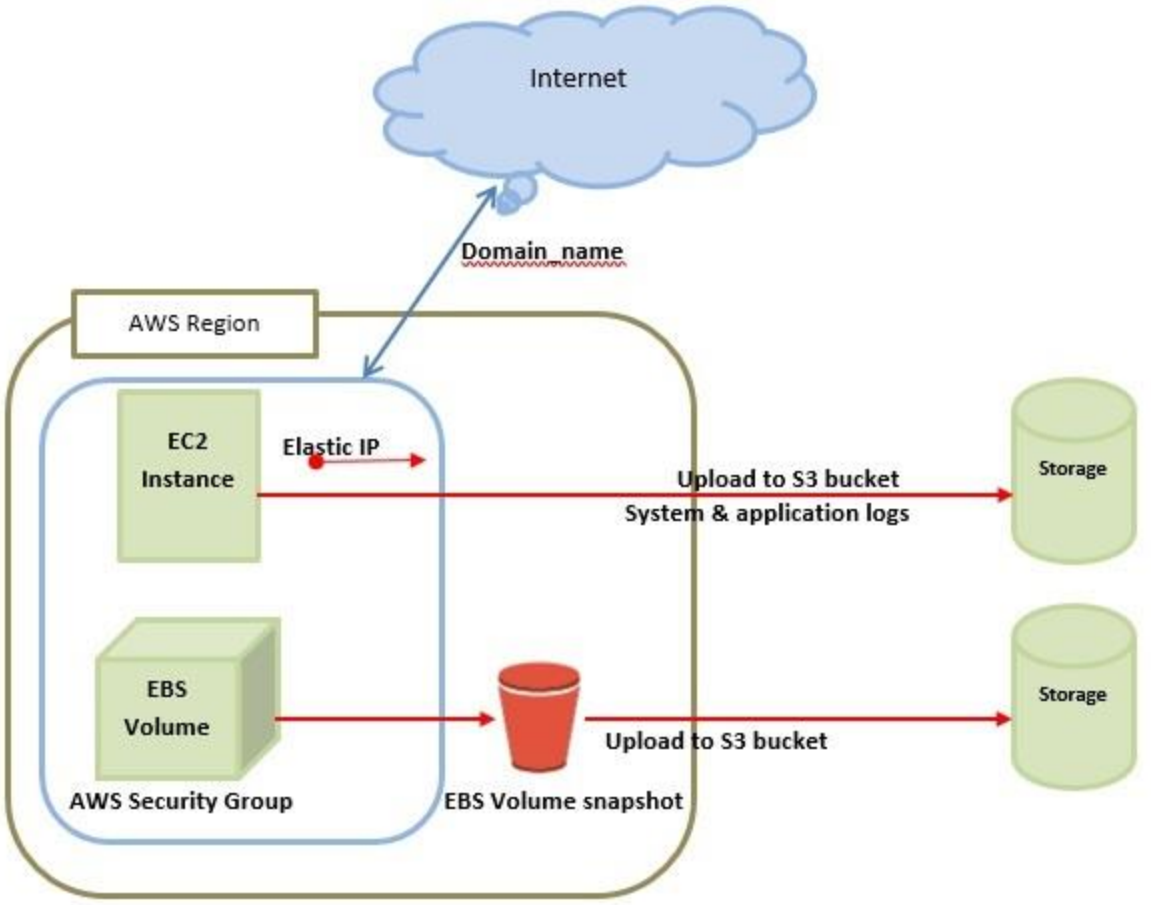

Fig 2: Amazon web service- Basic architecture.

- AWS LAMBDA Provides;
  - No server management
  - Developers have to pay only for the server space. This reduces the cost.
  - Serverless architectures are integrally scalable
  - Rapid deployments and updates are conceivable
  - Code can run closer to the end user, reducing latency

Simple Storage Service (S3)

Amazon Simple Storage Service (Amazon S3) is an object storage service that provides industry-leading scalability, data availability, security, and performance. Small, medium and big industries can use to store and protect any amount of data in websites [5-6], mobiles and enterprise applications etc.,

Elastic Transcoder

Amazon Elastic Transcoder is media transcoding in the cloud. This provides high scalability and cost-effective. This is used to play media files from their source format into the format which is adaptable to smartphones, tablets and PCS[7-8].

Kinesis

Because of Amazon Kinesis, collect, process and analyze real-time data is simple. Everything is done immediately and no need to wait.

METHODOLOGY

Here the following services has been taken for the risk mitigation.

Trigger AWS LAMBDA on S3

In order to avoid the unauthorized access, AWS LAMBDA with S3 can be utilized for event notification from Amazon

S3. During the object creation and deletion, Amazon S3 can send an event to a Lambda function. Configure notification settings on a bucket, and grant Amazon S3 permission to invoke a function on the function's resource-based permissions policy. S3 uses backup. When files are included in the source S3 bucket, it is automatically kept in another bucket for the purpose of security reason [9-10].

AWS S3 will trigger LAMBDA when a file is created or deleted into the bucket. AWS Lambda function will display an output in the form of console message that the file is uploaded. The user will be able to see the message in Cloud Watch logs. AWS S3 will trigger LAMBDA when a file is created or deleted into the bucket. AWS Lambda function will display an output in the form of console message that the file is uploaded. The user will be able to see the message in Cloud Watch logs.

Video Transcoding with AWS LAMBDA

Security Misconfigurations, Lack of Availability and Input of Malicious Contents in Video Transcoding is the reason to go with video transcoding with AWS LAMBDA. In traditional architecture manual configuration is needed for transcoding, which ends up in misconfiguration as well as injection of malicious contents and malwares into video files. If the video files are big, it leads to slow process. So, the solution for this risk is using AMAZON LAMBDA with Elastic Transcoder. LAMBDA automatically converts the file to multiple formats without manual intervention by mitigating all security risks. User uploads video file into S3. It triggers LAMBDA and sends an alert to user about the newly added file and it sends an event to Cloud Watch.

Kinesis with LAMBDA

It is used in the absence of Error Handling and Speed in Event Stream Data Processing. Kinesis reduces the workload and expenses as the user has to use the complex software and tons of infrastructure. The problem raised with Kinesis is that it reads data only up to a particular limit and the errors are not handled properly. The mitigation strategy used for the problem against Kinesis is to implement AWS LAMBDA. LAMBDA works as an event receiver, and it will read a large number of records. Errors are handled using LAMBDA and the error count will give a lot of insight into what is going on. Create data stream with kinesis and create an **IAM** (Identity and Access Management) role. Create a LAMBDA function, which will trigger Kinesis. LAMBDA will monitor the function, and handles the errors and it tells us everything about the execution of our function.

Result and Discussion

Fig 3 shows the aws management console. Fig 4 depicts the trigger. Fig 5 shows how the backup was done. Fig 6 displays about the video transcoder. This system avoid the risk by using the trigger, backup and make the video transcoder to be effective.

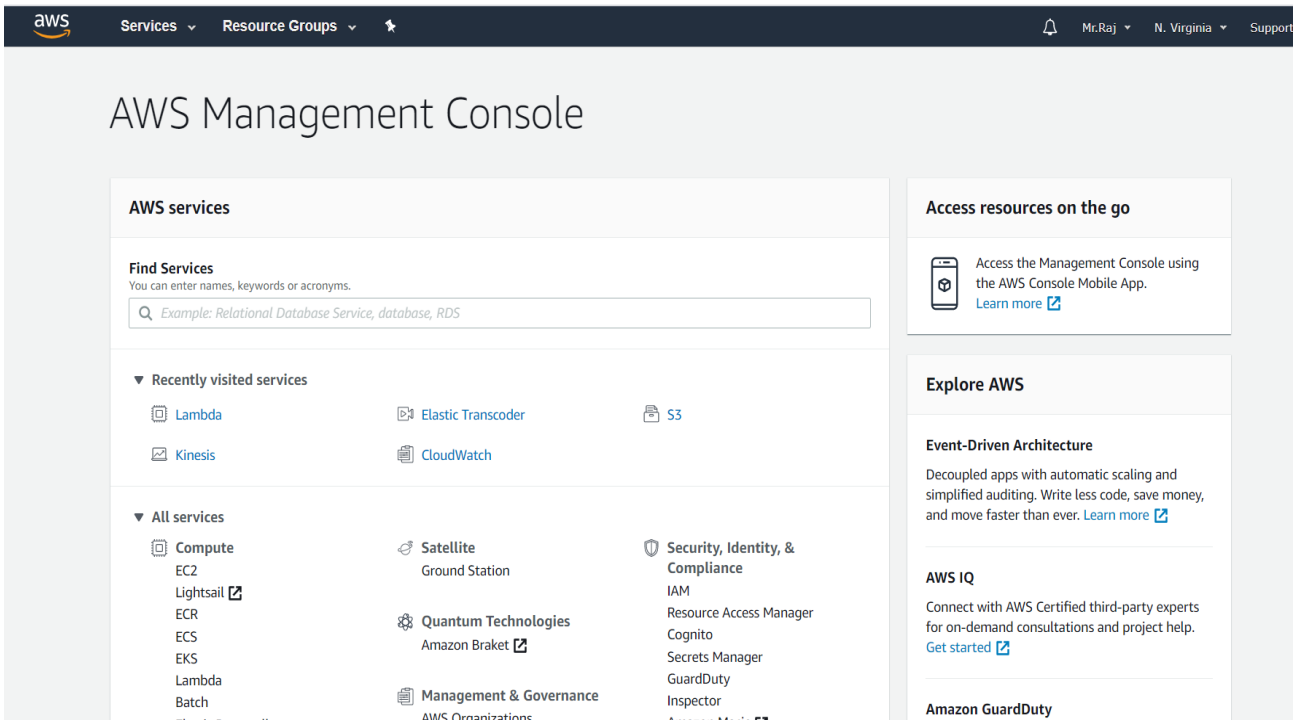

Fig. 3: AWS console

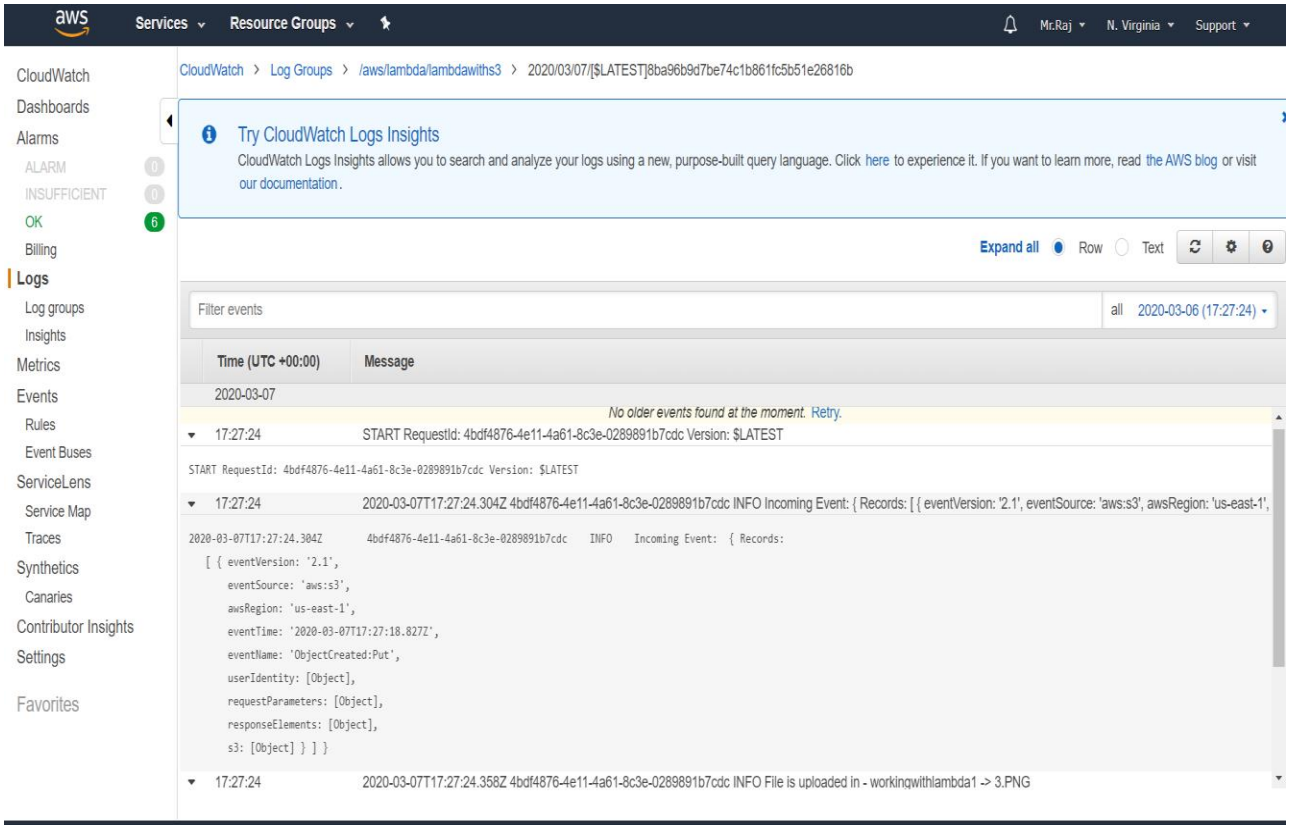

Fig 4. Notification send to the user using trigger.

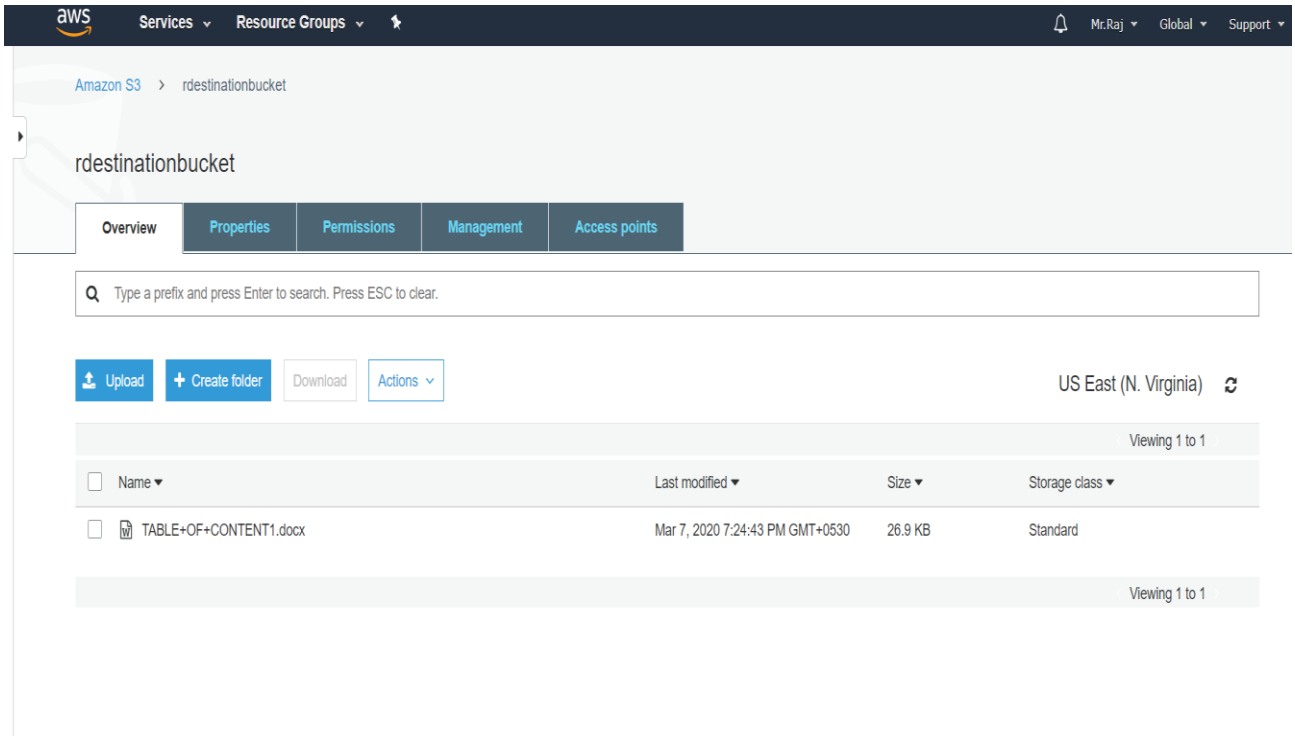

Fig 5. Backup

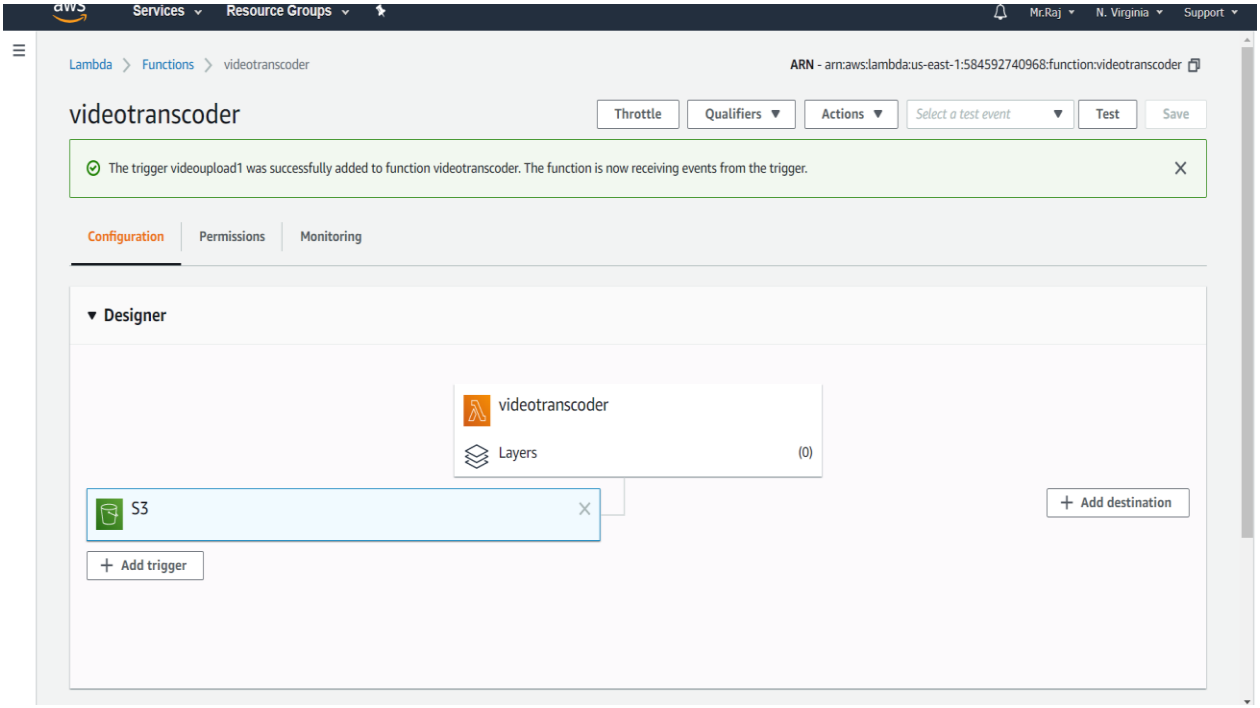

Fig 6: S3 trigger for video transcoder

Conclusion

Many organizations are adopting the serverless architecture. By integrating the AWS LAMBDA with traditional cloud architecture, there are many existing security problems are avoided. security misconfigurations, lack of availability and input of malicious contents in video transcoding, errors in data processing can be mitigated. Which results in enhancing the securiting and also improve the performance.

References

https://www.guru99.com/what-is-cloud-computing-with-example.html

www.tutorialpoint.com

Arokia Paul Rajan, R.Service request scheduling based on quantification principle using conjoint analysis and Z-score in cloud, (2018) International Journal of Electrical and Computer Engineering, 8 (2), pp. 1238-1246.

Barroso, L.A., Clidaras, J., Hölzle, U. The datacenter as a computer: An introduction to the design of warehouse-scale machines, second edition (2013) Synthesis Lectures on Computer Architecture, 24, pp. 1-156.

Van Eyk, E., Toader, L., Talluri, S., Versluis, L., Uta, A., Iosup, A. Serverless is More: From PaaS to Present Cloud Computing (2018) IEEE Internet Computing, 22 (5), art. no. 8481652, pp. 8-17.

Baldini, I., Castro, P., Chang, K., Cheng, P., Fink, S., Ishakian, V., Mitchell, N., (...), Suter, P.

Serverless computing: Current trends and open problems, (2017) Research Advances in Cloud Computing, pp. 1-20.

Saha, A., Jindal, S., EMARS: Efficient Management and Allocation of Resources in Serverless, (2018) IEEE International Conference on Cloud Computing, CLOUD, 2018-July, art. no. 8457882, pp. 827-830.

Lynn, T., Rosati, P., Lejeune, A., Emeakaroha, V., A Preliminary Review of Enterprise Serverless Cloud Computing (Function-as-a-Service) Platforms, (2017) Proceedings of the International Conference on Cloud Computing Technology and Science, CloudCom, 2017-December, pp. 162-169.

McGrath, G., Brenner, P.R. Serverless Computing: Design, Implementation, and Performance (2017) Proceedings - IEEE 37th International Conference on Distributed Computing Systems Workshops, ICDCSW 2017, art. no. 7979855, pp. 405-410. Cited 142 times.

Castro, P., Ishakian, V., Muthusamy, V., Slominski, A. Serverless Programming (Function as a Service) (2017) Proceedings - International Conference on Distributed Computing Systems, art. no. 7980271, pp. 2658-2659.

Ibrahim, S., & Koksal, M. E. (2021). Realization of a fourth-order linear time-varying differential system with nonzero initial conditions by cascaded two second-order commutative pairs. Circuits, Systems, and Signal Processing, 40(6), 3107-3123.