

# Configuration Manual

MSc Research Project Cloud Computing

# Divya Balasaheb Thorat Student ID: x18191878

School of Computing National College of Ireland

Supervisor: Vikas Sahni

#### National College of Ireland Project Submission Sheet School of Computing



Student Name:	Divya Balasaheb Thorat
Student ID:	x18191878
Programme:	Cloud Computing
Year:	2020
Module:	MSc Research Project
Supervisor:	Vikas Sahni
Submission Due Date:	17/08/2020
Project Title:	Configuration Manual
Word Count:	1000
Page Count:	14

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

<u>ALL</u> internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

I agree to an electronic copy of my thesis being made publicly available on TRAP the National College of Ireland's Institutional Repository for consultation.

Signature:	
Date:	17th August 2020

#### PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).							
Attach a Moodle submission receipt of the online project submission, to							
each project (including multiple copies).							
You must ensure that you retain a HARD COPY of the project, both for							
your own reference and in case a project is lost or mislaid. It is not sufficient to keep							
a copy on computer.							

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only							
Signature:							
Date:							
Penalty Applied (if applicable):							

# Configuration Manual

## Divya Balasaheb Thorat x18191878

## 1 Introduction

## 1.1 Motivation of the document

This presented manual describes requirement, deployment instruction, and validation for MapReduce application with a genetic algorithm on the serverless platform, according to National College of Ireland's project module handbook. The purpose of the configuration manual to define the prior requirement to implement MapReduce application with a genetic algorithm on a serverless platform.

## 1.2 Config Module Structure

Section	Purpose
Review	This section gives the information about im-
	plementation summary and amazon services re-
	quired to implement the research question.
Configuration required for development	To implement the research question this model
	describes the setup description.
Deployment for solution	This section describes the implementing proced-
	ure of deployment.
Validations	This section describes the experiment we have
	performed to validate the research question

# 2 Basic Information

## 2.1 Objective

The research question aims to reduce the total completion time of map-reduce application on serverless infrastructure. lambda function has better scalability and low cold start delay, a combination of slow and fast storage gives a better performance along with low cost, so we have used AWS services Pu (Feb 2019).

## 2.2 Implementation Summary

This model shows the implementation summary of project.

- Initially, the s3 bucket is use to store the static and dynamic data, coordinator lambda function use for making the connection between the mapper phase and reducer phase to

process the data. To fetch data from s3 to lambda function, we need to give some permission to the IAM role. The IAM role permissions include S3 full access, Cloud watch full access, s3 read-only access, VPC full access, elastic file system full access, lambda role, lambda VPC access executive role, and cloud watch events full access. The S3 trigger event is use to trigger the lambda function when data is inserted into a bucket.

- In this project, the coordinator function takes the input and distribute the data to two mapper functions with some conditions according to the data size.

- Mapper function is invoked from the coordinator function and discover good answer for the available task with the help of a genetic algorithm. Each mapper's data will store into Redis in-memory data storage to provide the excellent grained elasticity Pu (Feb 2019).

- Then the invocation of reducer is perform, and aggregated output is stored in another s3 bucket. To provide security for the applications, user processes the data with VPC by configuring the lambda function And creating the VPC endpoint for AWS S3 service.

#### 2.3 Architecture prerequisite

This section explains the services required to implement the research question. we are using the AWS services as the lambda service provides the low latency and scalability Wang et al. (2018).

#### 2.3.1 Amazon Web Services console account

In this it is necessity to create the console account to access the services.

#### 2.3.2 AWS Lambda

This service enables the user to run the code without provisioning and monitoring the servers. We just have to pay for compute the time of the service. It also provides continuous scaling whenever it will require. In this project, lambda service is used to create the coordinator, mapper, and reducer function. Coordinator function is used to distribute the data across the mapper function, each mapper function is used to find out the best output solution with the help of a genetic algorithm, and reducer function is used to aggregate the output of the mapper function. configuration done with the role, trigger event, network topology, security groups, and python language with the latest version 3.8 has been added at run time <sup>1</sup> Giménez-Alventosa et al. (2019).

#### 2.3.3 AWS S3

<sup>&</sup>lt;sup>1</sup>https://aws.amazon.com/lambda/

<sup>&</sup>lt;sup>2</sup>https://aws.amazon.com/s3

#### 2.3.4 AWS VPC

To access internet in AWS Lambda and elastic cache it is necessary to create the virtual private network. subnet, security group and route table are added in  $vpc^3$ .

#### 2.3.5 AWS Cloud Watch

The AWS Cloud Watch service is used for analysing monitoring, error handling, analysing logs and so on. This service provides all the information required for executing Mapreduce tasks at a given time (<sup>4</sup>.

#### 2.3.6 AWS Elastic Cache

To store the intermediate data generated through mapper we have used the elastic cache for  $\mathrm{redis}^5$ .

### 2.4 Run time Programming language

The run programming language used is Python 3.8 to create the functions. In this PyGAD is an open-source python 3 libraries for implementing the genetic algorithm. PyGAD provides assistance for distinct types of operators such as mutation, cross over, and parent selection. It allows a distinct type of problem is optimized using the genetic algorithm by the fitness function.

# 3 Configuration required for development

## 3.1 Code Repository

The zip file of the code is attach.

## 3.2 Required Run Time Programming Languages

The AWS lambda service gives platform for running code written in multiple language but we have used the python 3.8. The below packages has been imported for implementing the research question:

- boto3 1.10.28
- redis 5.0.6
- csv -13.1.1
- random
- json

<sup>&</sup>lt;sup>3</sup>https://aws.amazon.com/vpc/

<sup>&</sup>lt;sup>4</sup>https://aws.amazon.com/cloudwatch)

<sup>&</sup>lt;sup>5</sup>https://aws.amazon.com/elasticache

#### 3.3 Creation of IAM role

Before creating Lambda function, it is necessary create the IAM role which has below policies attach to it as shown in figure 1. so, it help the lambda function to create the execution environment.

- AmazonElasticCacheFullAccess
- AmazonS3FullAccess
- CloudWatchFullAccess
- AmazonVPCFullAccess
- AWSLambdaVPCAccessExecutionRole
- AWSLambdaRole
- CloudWatchEventsFullAccess
- AWSLambdaENIManagementAccess

₹ F	<ul> <li>Permissions policies (10 policies applied)</li> </ul>									
Attach policies O Add inline po										
	Policy name 👻	Policy type 👻								
•	AWSLambdaFullAccess	AWS managed policy	×							
►	AmazonElastiCacheFullAccess	AWS managed policy	×							
►	AmazonS3FullAccess	AWS managed policy	×							
►	CloudWatchFullAccess	AWS managed policy	×							
•	i AmazonVPCFullAccess	AWS managed policy	×							
►	AmazonElasticFileSystemFullAccess	AWS managed policy	×							
•	AWSLambdaVPCAccessExecutionRole	AWS managed policy	×							
•	🚺 AWSLambdaRole	AWS managed policy	×							
►	AWSLambdaENIManagementAccess	AWS managed policy	×							
►	CloudWatchEventsFullAccess	AWS managed policy	×							

Figure 1: Policies attach to a IAM role

#### 3.4 Creating VPC

we are creating VPC (server vpc) with 192.168.0.0/16 ipv4 CIDR block as shown in figure 2.

- 1. create public subnet with 192.168.10.0/24 v4 CIDR block as shown in figure 3
- 2. create private subnet with 192.168.20.0/24 ipv4 CIDR block as shown in figure 4.
- 3. create internet gateway to assign the public subnet shown in figure 5.
- 4. create NAT gateway assign to assign private subnet as shown in figure 6
- 5. create public route table and assign public subnet to it along with internet gateway as shown in figure 7.

- 6. create private route table and assign private subnet to it along with nat gateway and vpc end point as shown in figure 8
- 7. create vpc endpoint as shown in figure 9
- create the security group which act as virtual firewall so it allows the incoming traffic and outgoing traffic. In this security group it allows the traffic from anywhere to Redis and other services with the help of HTTP, HTTPs, SSH, custom TCP, and All ICMP – Ipv4 protocol as shown in figure 10.

Create VPC Actions *	0 ÷ 0
Q Filter by tags and attributes or search by keyword K < 1	to 2 of 2 > >
Name v VPC ID A State v IPv4 CIDR IPv6 CIDR IPv6 CIDR IPv6 CIDR (Network Border Group) DHCP options set	Main Route
server_vpc vpc-0d9ae93ca7e3f9fdf available 192.168.0 dopt-ef/2eb895	rtb-09e956ce
vpc-3295614f available 172.31.0 dopt-ef2eb895	rtb-4f3a4931
	Þ
VPC: vpc-0d9ae93ca7e3f9fdf	
Description CIDR Blocks Flow Logs Tags	
VPC ID vpc-0d9ae93ca7e3/9fdf Tenancy default	
State available Default VPC No	
IPv4 CIDR 192.168.0.0/16 Classic link Disabled	
IPv6 CIDR - IPv6 CIDR (Network Border -	
Group)	
IPv6 Pool - DNS resolution Enabled	
Network ACL acl-02ccd6e509afc5e81 DNS hostnames Disabled	
DHCP options set dopt-ef2eb895 ClassicLink DNS Support Disabled	
Route table         rtb-09e956ceet2056f66         Owner         52562883429	

Figure 2: creation of a vpc

Creat	te subnet	Actions v								ť	•	0
Q, F	Filter by tags an	nd attributes or se	earch by keyword							K < 1 to 9 c	f9 > >	>
	Name	- Subnet ID	Ŧ	State -	VPC	Ŧ	IPv4 CIDR -	Available	IPv4 - IPv6 CIDR	Availability Zor	ne - Ava	ailab
	public_subne	et subnet-04a	a52e4c876dc96a8	available	vpc-0d9ae93ca7e3f9fdf		192.168.10.0/	243	-	us-east-1b	use	e1-åz
	private_sub	subnet-09e	e36f55e23b5cd23	available	vpc-0d9ae93ca7e3f9fdf		192.168.20.0/	245	-	us-east-1b	use	e1-az
	private_sub	subnet-047	65f3865bcd3474	available	vpc-0d9ae93ca7e3f9fdf		192.168.30.0/	249	-	us-east-1b	use	e1-az
		subnet-77a	a87011	available	vpc-3295614f		172.31.0.0/20	4091	-	us-east-1b	use	e1-az
		subnet-a52	2d8a84	available	vpc-3295614f		172.31.80.0/20	4091	-	us-east-1c	use	ş1
D	escription	Flow Logs	Route Table	Network A	CL Tags		Sharing					
		Subnet ID	subnet-04a52e4c876d	c96a8				State	available			
	Available I	Pv4 Addresses	243	ii I server_vhc				IPV4 CIDR	192.100.10.0/24			
	Av	ailability Zone	us-east-1b (use1-az1)				Network Bo	order Group	us-east-1			
		Route Table	rtb-0a7a8130a4d565fd	c   public-RT			N	etwork ACL	acl-02ccd6e509afc5e81			
	I	Default subnet	No				Auto-assign	public IPv4 address	No			
	Auto-assig	n IPv6 address	No					Outpost ID	-			
		Owner	525262883429									

Figure 3: creation of a public subnet

#### 3.5 create s3 bucket

creating the s3 bucket to store the incoming data 11.

Crea	te subnet	Actions *								Ð	¢ 0
Q,	Filter by tags an	nd attributes or	search by keyword							I< ≤ 1 to 9 of 9	> >
	Name	- Subnet I	D ~	State -	VPC	Ŧ	IPv4 CIDR -	Available	IPv4 + IPv6 CIDR	Availability Zone -	Availab
	public_subne	et subnet-04	1a52e4c876dc96a8	available	vpc-0d9ae93ca7e3f9fdf		192.168.10.0/	243		us-east-1b	use1-az
	private_sub	subnet-0	9e36f55e23b5cd23	available	vpc-0d9ae93ca7e3f9fdf		192.168.20.0/	245		us-east-1b	use1-az
	private_sub	subnet-0	1765f3865bcd3474	available	vpc-0d9ae93ca7e3f9fdf		192.168.30.0/	249	-	us-east-1b	use1-az
		subnet-7	7a87011	available	vpc-3295614f		172.31.0.0/20	4091	-	us-east-1b	use1-az
		subnet-a	52d8a84	avallable	vpc-3295614f		172.31.80.0/20	4091		us-east-1c	use <sub>1</sub>
Subn	et: subnet-09e	Flow Log	s Route Table	Network A	CL Tags		Sharing			-	
		Subnet IE	subnet-09e36f55e23b	5cd23				State	available		
		VPC	vpc-0d9ae93ca7e3f9fd	If   server_vpc				IPv4 CIDR	192.168.20.0/24		
	Available II	Pv4 Addresses	245					IPv6 CIDR	-		
	A	ailability Zone	us-east-1b (use1-az1)				Network B	order Group	us-east-1		
		Route Table	rtb-06071e21d6c85a0	22   private-RT			N	etwork ACL	acl-02ccd6e509afc5e81		
		Default subne	t No				Auto-assign	public IPv4	No		
	Auto accia	n IPv6 address	No					address			
	Auto-abbigi	Owne	525262883429					Capostib			

Figure 4: creation of a private subnet

igw-08fede8e78a821f67 / My-IGW									
Details Info									
Internet gateway ID D igw-08fede8e78a821f67	State	VPC ID vpc-0d9ae93ca7e3f9fdf   server_vpc	Owner 🗗 525262883429						

Figure 5: creation of a internet gateway

Create N	IAT Gatew	Actions	*											Ð	¢ (	3
Q Filter	r by tags an	d attributes or se	arch by keywo	rd									K < 1 to 2	of 2	> >	
Na	ame	• NAT Gatev	vay ID ,	Status	⊸ Sta	itus Message	-	Elastic IP Addres	s -	Private IP Addr	ess -	Network Interface Iv	VPC	s	Subnet	
		nat-0954d3	11c38 a	available	-			18.211.70.239		192.168.30.108		eni-0695e09c31a	vpc-0d9ae93ca7e	. s	ubnet-04	F7
M)	YNAT	nat-0d8b38	28937 a	available	-			52.86.39.165		192.168.20.191		eni-0c23c91bfb44	vpc-0d9ae93ca7e	. s	ubnet-08	le
4														_		Þ
NAT Gate	eway: nat-( tails	Monitoring	afc5 Ta	gs												
	N	AT Gateway ID	nat-0d8b382	8937c6afc5						Status	availabl	e				
	St	atus Message	-						Elast	ic IP Address	52.86.3	9.165				
	Priva	te IP Address	192.168.20.1	191				N	etwor	k Interface ID	eni-0c2	3c91bfb44f4dbf				
		VPC	vpc-0d9ae93	3ca7e3f9fdf   se	rver_vpc					Subnet	subnet-	09e36f55e23b5cd23   pri	vate_subnet			
		Created	July 28, 2020	0 at 1:53:23 PM	I UTC+5:30					Deleted	-					

Figure 6: creation of a NAT gateway

Create route table	Actions *					÷	¢ 0
Q Filter by tags and a	attributes or search by keyword				[4	< 1 to 4 of 4	> >
Name -	Route Table ID	Explicit subnet associatior	Edge associations	Main	VPC ID ~	Owner	
private-RT	rtb-06071e21d6c85a022	2 subnets	-	No	vpc-0d9ae93ca7e3f9fdf	525262883429	
	rtb-09e956ceef2056f66	-	-	Yes	vpc-0d9ae93ca7e3f9fdf	525262883429	
public-RT	rtb-0a7a8130a4d565fdc	subnet-04a52e4c876dc96a8	-	No	vpc-0d9ae93ca7e3f9fdf	525262883429	
	rtb-4f3a4931	-		Yes	vpc-3295614f	525262883429	
Route Table: rtb-0a7a	8130a4d565fdc Routes Subnet Asso View All rout	Edge Associations	Route Propagation	Tags			
Destination		Target		Status	Propagated		
192.168.0.0/16		local		active	No		
0.0.0/0		igw-08fede8e7	8a821f67	active	No		

Figure 7: creation of a public route table

Creat	e route table	Actions *							Ð	• •	2
Q, F	ilter by tags and a	attributes or searc	h by keyword						< 1 to 4 of 4	> >	
	Name -	Route Table I	ID 🔶	Explicit subr	net associatior	Edge associations	Main	VPC ID	Owner		
	private-RT	rtb-06071e210	d6c85a022	2 subnets		-	No	vpc-0d9ae93ca7e3f9fdf	525262883429		
		rtb-09e956cee	ef2056f66	-		-	Yes	vpc-0d9ae93ca7e3f9fdf	525262883429		
	public-RT	rtb-0a7a8130a	a4d565fdc	subnet-04a52	e4c876dc96a8	-	No	vpc-0d9ae93ca7e3f9fdf	525262883429		
		rtb-4f3a4931		-		-	Yes	vpc-3295614f	525262883429		
< Route	Table: rtb-0607 Bummary	Routes	Subnet Assoc View All route	ciations E	idge Associations	Route Propagation	Tags				
	estination				Target		Status	Propagated			
1	92.168.0.0/16				local		active	No			
р 5	I-63a5400a (con 2.216.0.0/15, 3.	n.amazonaws.us 5.16.0/21, 3.5.0.	s-east-1.s3, 54.23 0/20)	31.0.0/17,	vpce-0160ea64	40fa181c00	active	No			
0	.0.0.0/0				nat-0d8b38289	137c6afc5	active	No			

Figure 8: creation of a private route table

Endpoints > Create Endpoint		
Create Endpoint		
A VPC endpoint allows you to securely conner An interface endpoint is powered by PrivateL A gateway endpoint serves as a target for a r	ct your VPC to another service. ink, and uses an elastic network interface (ENI) as an entry point for traffic destined to the service. oute in your route table for traffic destined for the service.	
Service category	AWS services Find service by name Your AWS Marketplace services	
	•	ó
	Q search: s3 💿 Add filter	X K < 1 to 1 of 1 > >
	Service Name Owner Type	
	com.amazonaw Service Name amazon Gateway	
VPC*	pc-3296614f   C 0	

Figure 9: creation of a vpc endpoint

Security group name	Security g	roup ID 50833480f433c5f	Description	VPC ID
Owner	Inbound r	ules count	Outbound rules count	
<b>D</b> 525262883429	11 Permis	sion entries	1 Permission entry	
Inbound rules Outbou	nd rules Tags			
Inbound rules				Edit inbound rules
Туре	Protocol	Port range	Source	Description - optional
НТТР	ТСР	80	0.0.0/0	-
нттр	ТСР	80	::/0	-
SSH	ТСР	22	0.0.0/0	-
SSH	ТСР	22	::/0	-
Custom TCP	TCP	6379	192.168.10.0/24	-
Custom TCP	TCP	6379	0.0.0.0/0	-

Figure 10: creation of a security group

		Create bucket			×
1 Name and region	on (2) Configu	ure options	3 Set permissions	(4) Review	
Name and reg	ion				
Bucket name 🌘					
Enter DNS-co	npliant bucket name				
Region					
US East (N. Vi	rginia)			~	
Copy settings	from an existing bucket				
Select bucket	<i>optional)</i> 1 Buckets			~	
Create				Cancel	ext

Figure 11: creation of a s3 bucket

## 3.6 Creation of Lambda function

After creating a IAM role, creating the lambda function for mapper, coordinator and reducer.

- To create lambda function go to AWS lambda and click on create a function as shown in figure 12.
- After selecting a create function now select the blueprint as shown in figure 13.
- now in a basic information section choose the function name, run-time language and permission i.e. IAM role with all policies that is required. as shown in figure 14]
- After creating the project lambda function now edit the vpc section and add the network configuration as shown in figure 15
- when there is input data in s3 bucket it should be notified to lambda function for that we have to create the s3 trigger event in lambda function 16

mbda 〉 Functions		
Functions (10)		C Actions Create function
Q Filter by tags and attributes or search by keywo	ord	? < 1 > ©
Function name		Runtime $ abla$ Code Last size $ abla$ modified $ abla$

Figure 12: Creation of function

Create function Info Choose one of the following options to crea	te your fur	nction.	
Author from scratch Start with a simple Hello World example.	0	Use a blueprint O Build a Lambda application from sample code and configuration presets for common use cases.	Browse serverless app repository O Deploy a sample Lambda application from the AWS Serverless Application Repository.

Figure 13: Selection of blueprint

Basic information
Function name Enters a name that describes the purpose of your function. project
Use only letters, numbers, hyphens, or underscores with no spaces. Runtime Info Choose the language to use to write your function. Python 3.8
Permissions into Lambda will create an execution role with permission to upload logs to Amazon CloudWatch Logs. You can configure and modify permissions further when you add triggers.
Evecution role Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console. Create a new role with basic Lambda permissions O Use an existing role Create a new role from AWS policy templates
Existing role Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs. Lambdapermission Very the Lambdapermission role on the IAM console.

## Figure 14: Configuration details

/PC Thoose a VPC for your function to access.	
vpc-0d9ae93ca7e3f9fdf (192.168.0.0/16)	▼ C
Subnets	
elect the VPC subnets for Lambda to use to set up your VPC configuration.	
	• C
subnet-09e36f55e23b5cd23 (192.168.20.0/24) us-east-1b X	
Hamer private_babilet	
subnet-04a52e4c876dc96a8 (192.168.10.0/24) us-east-1b 🗙 Name: public_subnet	
subnet-04765f3865bcd3474 (192.168.30.0/24) us-east-1b × Name: private_subnet	
Security groups Thoose the VPC security groups for Lambda to use to set up your VPC configur	, ration. The table below shows the inbound and outbour
utes for the security groups that you choose.	
	<ul> <li>C</li> </ul>
sg-0660833480f433c5f (securityforvpc) 🗙	

### Figure 15: Network Configuration

		basic_setup	
		Eayers	
<b>53</b>			×
+ Add trigger			
S3			
inputasinitial am:aws:s3:::inputasinitial Event type: ObjectCreated	Notification name: 4f87cdf3_dfd4_4305_8aa4_b63	626266534	
eren eyper e bjeccereacea	110-110-110-110-110-010-010-010-010-000-000-000-000-000-000-000-000-000-000-000-000-000-000-000-000-000-000-000		



#### 3.7 Create Cache Memory

create the elastic cache memory for redis to store the intermediate data generated from mapper function.

- 1. select the cluster engine as redis as shown in figure 17
- 2. then configure a redis setting in which select the engine version, port, node type and number of replica as shown in figure 18
- 3. in advance setting need to select the subnet group, security group, backup and maintenance window as shown in figure19

Create your Amazon ElastiCache cluster					
Cluster engine	۲	Redis         In-memory data structure store used as database, cache and message broker. ElastiCache for Redis offers Multi-AZ with Auto-Failover and enhanced robustness.         Cluster Mode enabled			
		Memcached High-performance, distributed memory object caching system, intended for use in speeding up dynamic web applications.			

Figure 17: Redis Configuration

Redis settings		
Name		0
Description		0
Engine version compatibility	5.0.6	0
Port	6379	0
Parameter group	default.redis5.0	0
Node type	cache.t2.micro (0.5 GiB)	0
Number of replicas	1	0
Multi-AZ	0	0

Figure 18: Redis setting

## 4 Validation

create a lambda function and inside that function configure a mapper, coordinator and a reducer function. now prepare the code and deploy it into a function using a upload zip

<ul> <li>Advanced Redis settings</li> </ul>									
Advanced settings have common defaults set to give you the fastest way to get started. You can modify these now or after your cluster has been created.									
Subnet group	lambdatoredis (vpc-0d9ae93ca7e3f9fdf)	• 0							
Availability zones placement	No preference     Select zones	0							
Security									
Security groups	default (sg-0224ff9fd02526adb) 🖋	0							
Encryption at-rest		0							
Encryption in-transit		0							

Figure 19: Advance redis setting

file option. Now configure the network configuration inside vpc section. after creating all the services now test the configuration. Testing of the configuration done in two ways as shown below.

- first way is to create test inside a function as shown in a figure 20. Run the test and the results are displayed in figure 21. An error is thrown when there is no correct configuration.
- second way is to add the input in s3 bucket as shown in figure 22. so whenever there is input at s3 bucket s3 event trigger is happen. If there is proper connection between s3 trigger event and lambda function then it will execute automatically, the output is displayed in cloud watch service as shown in figure 23.

Configure test event	×
A function can have up to 10 test events. The events are persisted so you can switch to another computer or web browse and test your function with the same events.	er
• Create new test event	
Edit saved test events	
Event template	
hello-world	•
Event name	
MyEventName	
<pre>1* [{ 2     "key1": "value1", 3     "key2": "value2", 4     "key3": "value3" 5 ]</pre>	

Figure 20: Creation of test event

■ Execution Result × ⊕			
Execution results	Status: Succeeded	Max Memory Used: 80 MB	Time: 259.87 ms
Response: "County,Category\r\nAlbany,Animal\r\nAlbany,Animal\r\nAlbany,Animal\r\nAlbany,Animal\r\nAlbany,Animal\r\nAlbany,Animal Response: TO:	al\r\nAlbany,Anim	al\r\nAlbany,Natural (	`ommunity\r\n
request LD: "ceb39ca9-fc9e-4c79-81a3-6b8daf180945"			
Function logs. START RequestId: ceb39ca9-fc9e-4c79-81a3-6b8daf180945 Version: \$LATEST FileSize = 0.0150241851806664062 MB chunk size to small (d butes) changing to 1024 butes			
chunk size to large (024 bytes), changing to 102 bytes Generation 0 CPU best fit: 1 generation: 3 _ fitness of Population: 0.3932160000000007			
EHD RequestId: ceb39ca9-fc9e-4c79-81a3-6b8daf180945 REPORT RequestId: ceb39ca9-fc9e-4c79-81a3-6b8daf180945 Duration: 259.87 ms Billed Duration: 300 ms Memory Size: 832	MB Max Memory Us	ed: 80 MB Init Durati	on: 437.08 m



aws-trigger1							
Overview Properties	Permissions	Management	Access points				
<b>Q</b> Type a prefix and press Enter to	o search. Press ESC to cle:	ar.					
1 Upload + Create folder	Download Actions	×				US East (N. Virgir	nia) <i>C</i>
						Viewing 1	to 3
Name 🔻			Las	st modified 🔻	Size 💌	Storage class 💌	
RedditNews.csv			Au	g 10, 2020 12:23:04 AM GMT+0530	8.7 MB	Standard	
country_anima.csv			Au	g 12, 2020 6:04:08 PM GMT+0530	15.4 KB	Standard	
ABC.csv			Au	g 10, 2020 8:24:43 PM GMT+0530	3.2 MB	Standard	
						Viewing	to 3

Figure 22: S3 Trigger event

Cloud	Watch > CloudWatch Logs > Log gro	bups       > /aws/lambda/redis       > 2020/08/12/[\$LATEST]bb55f6dd2a8e4b308eac4f14d3828fc7       Switch to the original interface
Lo	g events	C Actions  Create Metric Filter
C	Filter events	Clear 1m 30m 1h 12h Custom 🗐 📀
Þ	Timestamp	Message
		There are older events to load. Load more.
•	2020-08-12T19:34:53.254+05:30	START RequestId: ceb39ca9-fc9e-4c79-81a3-6b8daf180945 Version: \$LATEST
- F	2020-08-12T19:34:53.476+05:30	FileSize = 0.015024185180664062 MB
Þ	2020-08-12T19:34:53.494+05:30	chunk size to small (0 bytes), changing to 1024 bytes
Þ	2020-08-12T19:34:53.494+05:30	chunk size to large (1024 bytes), changing to 460 bytes
Þ	2020-08-12T19:34:53.494+05:30	Generation 0 CPU best fit: 1
•	2020-08-12T19:34:53.512+05:30	generation: 3 , fitness of Population: 0.3932160000000007
Þ	2020-08-12T19:34:53.514+05:30	END RequestId: ceb39ca9-fc9e-4c79-81a3-6b8daf180945
- Þ	2020-08-12T19:34:53.515+05:30	REPORT RequestId: ceb39ca9-fc9e-4c79-81a3-6b8daf180945 Duration: 259.87 ms Billed Duration: 300 ms Memory Size: 832 MB Max Memory Used: 8
		No newer events at this moment. Auto retry paused. Resume

Figure 23: Cloud Watch logs

## References

- Giménez-Alventosa, V., Moltó, G. and Caballer, M. (2019). A framework and a performance assessment for serverless MapReduce on AWS lambda, 97: 259–274. journal Impact Factor: 4.639. URL: http://www.sciencedirect.com/science/article/pii/S0167739X18325172
- Pu, Q. (Feb 2019). Shuffling, fast and slow: Scalable analytics on serverless infrastructure, p. 15.
  URL: https://www.usenix.org/system/files/nsdi19-pu.pdf
- Wang, L., Li, M., Zhang, Y., Ristenpart, T. and Swift, M. (2018). Peeking behind the curtains of serverless platforms, p. 13. URL: https://www.usenix.org/system/files/conference/atc18/atc18-wang-liang.pdf