

Configure and Connect a MySQL Database Instance with a Web Server

Project 1

DESCRIPTION

You are working as a database administrator for an IT firm. You have been asked to create a new database instance on AWS cloud and connect it with the employee management portal hosted on a web server.

Background of the problem statement:

Your organization wants to deploy a new multi-tier application. The application will take live inputs from the employees and it will be hosted on a web server running on the AWS cloud.

The development team has asked you to set up the web server and configure it to scale automatically in cases of a traffic surge, to make the application highly available. They have also asked you to take the inputs from the employees and store them securely in the database.

You must use the following:

- Create a Database Instance with the following specifications:
 - Database creation method: Standard Create
 - Engine: MySQL
 - Database Instance size: db.t2.micro
- Create an EC2 Instance with the following specifications:
 - AMI: Amazon Linux
 - Region: Use only US East (N Virginia), us-east-1, and us-east-2
 - Instance types: t2.micro and t3.micro
 - Allowed EBS types: GP2 and Standard

The following requirements should be met:

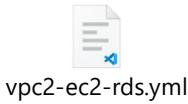
- Follow the above-mentioned specifications
- Make sure that the Availability Zone is similar throughout the instances and volumes
- Ensure that the server scales automatically and the traffic is optimally routed among the scaled servers
- Document the step-by-step process involved in completing this task

Summary

I went through the manual creation of each resource step by step and completed the project as requested. But I decided later to use “AWS CloudFormation” feature for the final project submission in order to simplify or avoid the repeated creation of AWS resources. The template describes the resources that I want to provision: deploys a VPC, with a pair of: public, private, and data subnets spread across two Availability Zones (us-east-1a and us-east-1b).

It deploys an Internet Gateway, with a default route on the public subnets and NAT gateway and route for the private subnets. Associate subnets with the Route Tables, and Created a network interface with IP in the subnet that was created. Assign an elastic IP to the network interface created. EC2 and RDS instances were also created using the same template. An EC2 security group is configured to allow inbound access on port 22 from the CIDR IP address range that the user specifies.

Here is the embedded template that was used for the project.



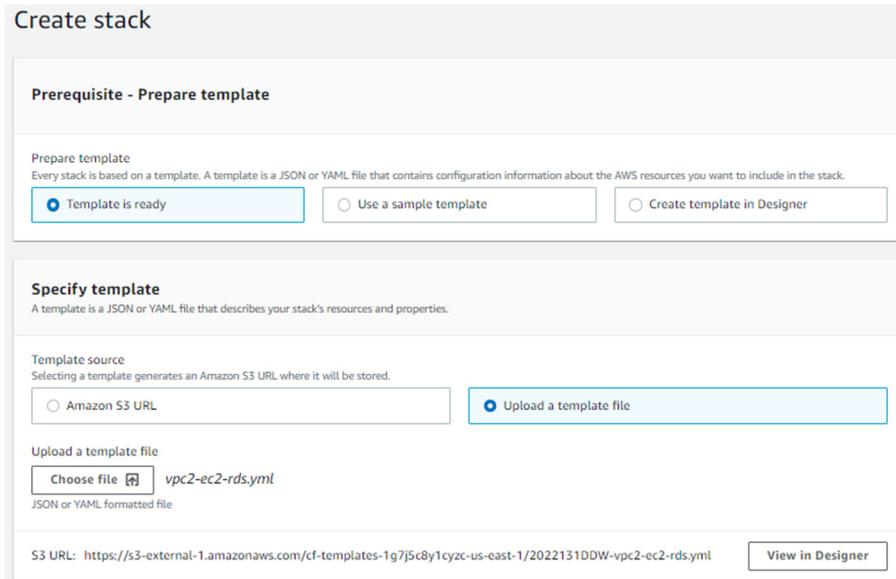
The following steps were taken in the attached “.yml” template file:

- Used Metadata ParameterGroups in order to organize each configuration type for use in the AWS console.
 - Network Configuration
 - EC2 Instances Configuration
 - RDS Instance Configuration
- Define Parameters that the user must specify to launch the product. For each parameter, the template includes a description and constraints that must be met by the value typed.
 - Entry CIDR for VPC and all subnets
 - Entry EC2 and RDS instance parameters
- List all resources used: Networking, An EC2 instance running Amazon Linux and a security group that allows SSH access to the instance, and RDS instance
- Created WebServer and RDSServer instances, and created security groups
- Attached UserDate on the WebServer using the following commands:

```
UserData:  
Fn::Base64:  
!Sub |  
#!/bin/bash -x  
sudo yum update -y  
sudo amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2  
sudo yum install -y httpd  
sudo systemctl start httpd  
sudo systemctl enable httpd  
sudo usermod -a -G apache ec2-user  
groups  
ec2-user adm wheel apache systemd-journal
```

```
sudo chown -R ec2-user:apache /var/www
sudo chmod 2775 /var/www
find /var/www -type d -exec sudo chmod 2775 {} \;
find /var/www -type f -exec sudo chmod 0664 {} \;
cd /var/www
mkdir inc
aws s3 cp s3://my-s1-webserver/dbinfo.inc /var/www/inc/
aws s3 cp s3://my-s1-webserver/SamplePage.php /var/www/html/
```

- Then I used the template file to create the stack.



Network, EC2, and RDS Configuration parameters are entered as below.

Network Configuration

VPC CIDR

Please enter the IP range (CIDR notation) for this VPC

PublicSubnet1CIDR

Please enter the IP range (CIDR notation) for the public subnet in the first Availability Zone

PublicSubnet2CIDR

Please enter the IP range (CIDR notation) for the public subnet in the second Availability Zone

PrivateSubnet1CIDR

Please enter the IP range (CIDR notation) for the private subnet in the first Availability Zone

PrivateSubnet2CIDR

Please enter the IP range (CIDR notation) for the private subnet in the second Availability Zone

DataSubnet1CIDR

Please enter the IP range (CIDR notation) for the data subnet in the first Availability Zone

DataSubnet2CIDR

Please enter the IP range (CIDR notation) for the data subnet in the second Availability Zone

EC2 Instances Configuration

Environment

Application environment for which this network is being created. e.g. Development/Production.



InstanceName

InstanceType

WebServer EC2 instance type



InstanceAMI

AMI for use with the EC2 instances



InstancePublicIP

Specifies whether to launch instances with public IP addresses in your VPC.



KeyName

Name of an existing EC2 KeyPair to enable SSH access to the instances



RDS Instance Configuration

DBInstanceId

The RDS DB Instance Identifier

DBName

The RDS DB Instance Name

DBEngine

The RDS DB Engine Type

DBUserName

The RDS DB Instance UserName

DBUserPass

The RDS DB Instance Password

DBStorage

The RDS DB Size in GB

DBInstanceClass

The RDS DB Instance Type

AWS Console Verification of the resources created.

- After successful completion of the stack creation, I verified all the resources to make sure they were correctly created.
- I show a few snapshots from the console:

VPC

Your VPCs (3) Info											
<input type="button" value="Actions"/> Create VPC < 1 >											
	Name	VPC ID	State	IPv4 CIDR	IPv6 ...	DHCP options set	Main route table	Main network ACL	Tenancy	Default V	
<input type="checkbox"/>	cfn-stack-VPC	vpc-0b70fff53a92cea68	Available	10.22.0.0/16	-	dopt-0867c0f11734ad87	rtb-0bcfbe1aa397416cb	acl-080e9fd1217abc3d	Default	No	
<input type="checkbox"/>	proot	vpc-0ca51c6f876bf1c20	Available	172.31.0.0/16	-	dopt-0867c0f11734ad87	rtb-0b610495adaab3167	acl-0c87fb82be3615dac	Default	Yes	

Subnets

Subnets (1/8) Info											
<input type="button" value="Actions"/> < 1 >											
	Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 ...	Available IPv4 addresses	Availability Z			
<input checked="" type="checkbox"/>	cfn-stack-public-subnet-az2	subnet-0b1eaddd1a0a1f41f	Available	vpc-0b70fff53a92cea68 cfn-stack-VPC	10.22.1.0/24	-	251	us-east-1b			
<input type="checkbox"/>	cfn-stack-public-subnet-az1	subnet-0e846c4e48b6cf73a	Available	vpc-0b70fff53a92cea68 cfn-stack-VPC	10.22.0.0/24	-	249	us-east-1a			
<input type="checkbox"/>	cfn-stack-private-subnet-az2	subnet-0ea225912f365b144	Available	vpc-0b70fff53a92cea68 cfn-stack-VPC	10.22.3.0/24	-	251	us-east-1b			
<input type="checkbox"/>	cfn-stack-private-subnet-az1	subnet-04fa4674fc8580977	Available	vpc-0b70fff53a92cea68 cfn-stack-VPC	10.22.2.0/24	-	251	us-east-1a			
<input type="checkbox"/>	cfn-stack-data-subnet-az2	subnet-06e046cce58cf2a56	Available	vpc-0b70fff53a92cea68 cfn-stack-VPC	10.22.5.0/24	-	251	us-east-1b			
<input type="checkbox"/>	cfn-stack-data-subnet-az1	subnet-09e60c0f565afa0eb	Available	vpc-0b70fff53a92cea68 cfn-stack-VPC	10.22.4.0/24	-	250	us-east-1a			

Route Tables: Created one for 2 public subnets and two for each of private subnets

Route tables (11) Info						
<input type="button" value="Actions"/> < 1 >						
	Name	Route table ID	Explicit subnet associations	Ed...	M... ▾	VPC
<input type="checkbox"/>	cfn-stack-public-rtb	rtb-02b81fbda3e4d1e7e	2 subnets	-	No	vpc-0b70fff53a92cea68 cfn-stack-VPC
<input type="checkbox"/>	cfn-stack-private-rtb-az2	rtb-0b3c3b15ddec9a278	2 subnets	-	No	vpc-0b70fff53a92cea68 cfn-stack-VPC
<input type="checkbox"/>	cfn-stack-private-rtb-az1	rtb-0a090efcb49e77a5e	2 subnets	-	No	vpc-0b70fff53a92cea68 cfn-stack-VPC

Subnets (1/8) [Info](#)

Subnets (1/8) Info						
<input type="button" value="Actions"/> < 1 >						
	Name	Subnet ID	State	VPC	IPv4 CIDR	
<input checked="" type="checkbox"/>	cfn-stack-public-subnet-az2	subnet-0b1eaddd1a0a1f41f	Available	vpc-0b70fff53a92cea6...	10.22.1.0/24	
<input type="checkbox"/>	cfn-stack-public-subnet-az1	subnet-0e846c4e48b6...	Available	vpc-0b70fff53a92cea6...	10.22.0.0/24	
<input type="checkbox"/>	cfn-stack-private-subnet-az2	subnet-0ea225912f36...	Available	vpc-0b70fff53a92cea6...	10.22.3.0/24	
<input type="checkbox"/>	cfn-stack-private-subnet-az1	subnet-04fa4674fc85...	Available	vpc-0b70fff53a92cea6...	10.22.2.0/24	

subnet-0b1eaddd1a0a1f41f / cfn-stack-public-subnet-az2

Details	Flow logs	Route table	Network ACL	CIDR reservations	Sharing	Tags
<small>(1) You can now check network connectivity with Reachability Analyzer</small>						
Route table: rtb-02b81fbda3e4d1e7e / cfn-stack-public-rtb						
Routes (2)						
<input type="button" value="Actions"/> < 1 >						
<input type="button" value="Filter routes"/>						
Destination		Target				
10.22.0.0/16		local				
0.0.0.0/0		igw-0317b6d0b26c8712c				

Internet Gateway attached to VPC

Internet gateways (1/1) Info					
<input type="text"/> Filter internet gateways					
search: cf X		Clear filters			
✓	Name	Internet gateway ID	State	VPC ID	Owner
<input checked="" type="checkbox"/>	cfn-stack-InternetGateway	igw-0317b6d0b26c8712c	<input checked="" type="checkbox"/> Attached	vpc-0b70fff53a92cea68 cfn-stack-VPC	046672410778

igw-0317b6d0b26c8712c / cfn-stack-InternetGateway

Details | Tags

Details

Internet gateway ID igw-0317b6d0b26c8712c	State Attached	VPC ID vpc-0b70fff53a92cea68 cfn-stack-VPC
--	---	---

NAT Gateway – attached to public subnet on az1

NAT gateways (1/1) Info										Actions	Create NAT gateway	
										Filter NAT gateways	Actions	Create NAT gateway
Name	NAT gateway ID	Connection type	State	State message	Elastic IP address	Private IP address	Network interface ID	VPC	Subnet			
cfn-stack	nat-0acaef85a9cac872	Public	Available	—	52.1.167.126	10.22.0.171	eni-01b0a2966ed0b1c...	vpc-0b70ff53a92cea6...	subnet-0e846c4e48b6...			

nat-0acae98f5a9cac872 / cfn-stack

[Details](#) | [Monitoring](#) | [Tags](#)

Details

NAT gateway ID	Connectivity type	State	State message
nat-0cae98f5a9cac872	Public	Available	-
NAT gateway ARN	Elastic IP address	Private IP address	Network interface ID
arn:aws:ec2:us-east-1:046672410778:natgateway/nat-0cae98f5a9cac872	52.1.167.126	10.22.0.171	eni-01b0a2966ed0b1c46
VPC	Subnet	Created	Deleted
vpc-0b70fff53a92cea68 / cfn-stack-VPC	subnet-0e846c4e48b6cf73a / cfn-stack-public-subnet-az1	 Tuesday, May 10, 2022, 22:06:37 EDT	-

EC2 Instance

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name	Monitoring	Launch Time	Security Groups
cfn-stack	i-0319070750d0f9a0d	t2.micro	us-east-1a	running	2/2 checks ...	None	ec2-3-87-155-27.compute-1.amazonaws.com	3.87.155.27	-	abebe_key1	disabled	May 10, 2022 at 10:06:05 P...	cfn-stack-WebSe...

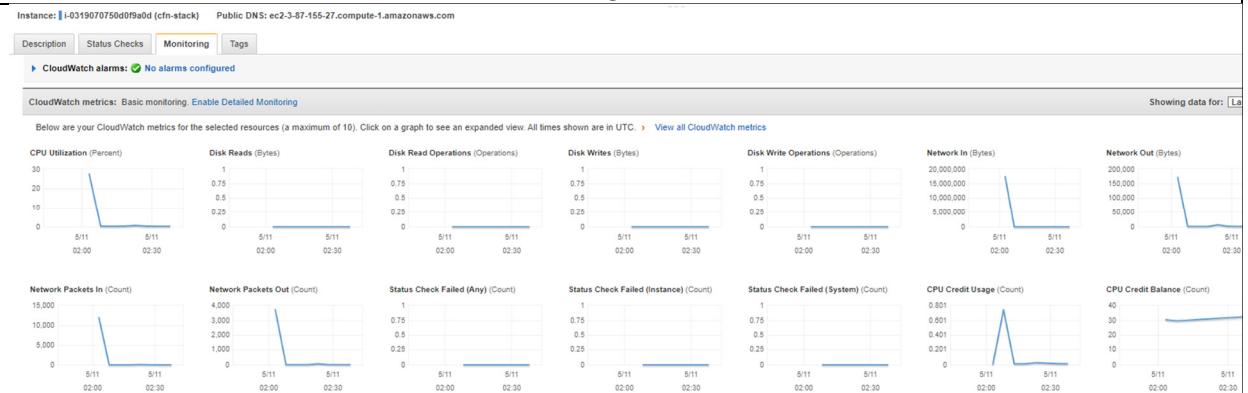
Instance: i-0319070750d0f9a0d (cfn-stack) Public DNS: ec2-3-87-155-27.compute-1.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID: i-0319070750d0f9a0d
 Instance state: running
 Instance type: t2.micro
 Finding: Option to AWS Compute Optimizer for recommendations. Learn more
 Private DNS: ip-10-22-0-100.ec2.internal
 Private IP: 10.22.0.190
 Secondary private IP:
 VPC ID: vpc-0b70ff3a92eaa58 (cfn-stack-VPC)
 Platform: Amazon Linux
 Platform details: Linux/UNIX
 Usage operation: RunInstances
 Source/dest check: True
 T2/T3 Unlimited: Disabled
 EBS-optimized: False
 Root device type: ebs
 Root device: /dev/xvda
 Block devices: /dev/xvda
 Elastic Graphics ID: -
 Elastic Inference accelerator ID: -
 Capacity Reservation: -
 Capacity Reservation Settings: Open
 Outpost AM: -

Public DNS (IPv4): ec2-3-87-155-27.compute-1.amazonaws.com
 IPv4 Public IP: 3.87.155.27
 IPv6 IP: -
 Elastic IP: -
 Availability zone: us-east-1a
 Security groups: cfn-stack-WebSeGrp-CNWLZLZ0T50K, view inbound rules, view outbound rules
 Scheduled events: No scheduled events
 AMI ID: amzn2ami-kernel-5.10-hvm-2.0.20220426.0-x86_64-gp2 (ami-0022774911c1d690)
 Subnet ID: subnet-0e345c4e43bd0cf73a (cfn-stack-public-subnet-ez1)
 Network interfaces: eth0
 IAM role: -
 Key pair name: abebe_key1
 Owner: 046672410778
 Launch time: May 10, 2022 at 10:06:05 PM UTC-4 (less than one hour)
 Termination protection: False
 Lifecycle: normal
 Monitoring: basic
 Alarm status: None
 Kernel ID: -
 RAM disk ID: -
 Nitro Enclaves: Disabled
 Placement group: -
 Partition number: -
 Virtualization: hvm
 Reservation: r-0bd9258f17d0d495f
 AMI launch index: 0
 Tenancy: default
 Host ID: -
 Host resource group name: -
 Affinity: -
 State transition reason: -
 State transition reason message: -
 Stop - Hibernation behavior: Disabled
 Number of vCPUs: 1

EC2 Instance CloudWatch Metrics Basic Monitoring



EC2 Volume of 8 GiB and gp2 type

Create Volume Actions

search: 61f Add filter

Name	Volume ID	Size	Volume Type	IOPS	Thru...	Snapshot	Created	Availability Zone	State	Alarm Status	Attachment
vol-0a98c5c42a1aad61f	8 GiB	gp2	100	-	snap-08ccb15f1c8eb5387	May 10, 2022 at 10:06:06 PM UTC-4	us-east-1a	in-use	None	i-0319070...	

Volumes: vol-0a98c5c42a1aad61f

Description Status Checks Monitoring Tags

Volume ID: vol-0a98c5c42a1aad61f
 Alarm status: None
 Snapshot: snap-08ccb15f1c8eb5387
 Availability Zone: us-east-1a
 Encryption: Not Encrypted
 KMS Key ID: -
 KMS Key Aliases: -
 KMS Key ARN: -
 Throughput (MB/s): -

Outputs ARN: -
 Size: 8 GiB
 Created: May 10, 2022 at 10:06:06 PM UTC-4
 State: in-use
 Attachment information: i-0319070750d0f9a0d (cfn-stack)/dev/xvda (attached)
 Volume type: gp2
 Product codes: -
 IOPS: 100
 Multi-Attach Enabled: No

Security groups

Security Groups (2) Info

	Name	Security group ID	Security group name	VPC ID	Description
<input type="checkbox"/>	cfn-stack	sg-0b6e4d5dc1fc841	cfn-stack-WebSecGrp-CNW6LLZ0T5OK	vpc-0b70fff53a92cea68	Enable SSH & HTTPD access via port 22 & 80 respectively
<input type="checkbox"/>	cfn-stack	sg-0fd103d297ac0c354	cfn-stack-DBSecGrp-F60X1QZZWN64	vpc-0b70fff53a92cea68	Frontend Access - Enable EC2 to access RDS(MySQL) access via port 3306

Elastic IP addresses (1/1)

<input checked="" type="checkbox"/>	Name	Allocated IPv4 ...	Type	Allocation ID
<input checked="" type="checkbox"/>	cfn-stack	52.1.167.126	Public IP	eipalloc-07e5999b1516ddac4

Subnet Groups for RDS Instance

RDS > Subnet groups

Subnet groups (5)				
<input type="text"/> cfn				
	Name	Description	Status	VPC
<input type="checkbox"/>	cfn-stack-dbsubnetgrp-tfm1jar1yiaa	The RDS DB Instance Identifier	Complete	vpc-0b70fff53a92cea68
<input type="checkbox"/>	cfn-stack-rdssubnetgroup-13pd5645bsugf	RDS DB Subnet group.	Complete	vpc-0b70fff53a92cea68

RDS Instance

RDS > Databases > cfn-rds-inst01

cfn-rds-inst01 Modify

Summary

DB identifier cfn-rds-inst01	CPU <div style="width: 4.14%;">4.14%</div>	Status Available	Class db.t2.micro
Role Instance	Current activity 0 Connections	Engine MySQL Community	Region & AZ us-east-1a

Connectivity & security Monitoring | Logs & events | Configuration | Maintenance & backups | Tags

Endpoint & port	Networking	Security
Endpoint cfn-rds-inst01.cyzpvqfcc7eq.us-east-1.rds.amazonaws.com	Availability Zone us-east-1a	VPC security groups cfn-stack-DBSecGrp-F60X1QZZWN64 (sg-0fd103d297ac0c354) Active
Port 3306	VPC cfn-stack-VPC (vpc-0b70fff53a92cea68)	Public accessibility No
	Subnet group cfn-stack-dbsubnetgrp-tfm1jar1yiaa	Certificate authority rds-ca-2019
	Subnets subnet-06e046cce58cfaf256 subnet-09e60c0f565afa0eb	Certificate authority date August 22, 2024, 01:08 (UTC±1:08)
	Network type IPv4	

RDS Instance Configuration			
Instance			
Configuration	Instance class	Storage	Performance Insights
DB instance ID cfn-rds-inst01	Instance class db.t2.micro	Encryption Not enabled	Performance Insights enabled No
Engine version 8.0.28	vCPU 1	Storage type Magnetic	Database activity stream
DB name wpdb01	RAM 1 GB	Provisioned IOPS -	Status Stopped
License model General Public License	Availability	Storage 20 GiB	
Option groups default:mysql-8-0 In sync	Master username dbuser	Storage autoscaling Disabled	
Amazon Resource Name (ARN) arn:aws:rds:us-east-1:046672410778:db:cfn-rds-inst01	IAM DB authentication Not enabled		
Resource ID db-IQW3CPWJPAH7VOJNGZDGGI52IU	Multi-AZ No		
Created time Tue May 10 2022 22:13:31 GMT-0400 (Eastern Daylight Time)	Secondary Zone -		
Parameter group default.mysql8.0 In sync			
Deletion protection Disabled			
AWS CLI			
<ul style="list-style-type: none"> AWS CLI was used to verify the installation of each application provided from UserData Downloaded the S3 files: dbinfo.inc and SamplePage.php to the ec2 instances to the folders /var/www/inc and /var/www/html folders respectively 			
<pre>[ec2-user@ip-10-22-0-190 www]\$ ls -alt total 0 drwxrwsr-x 5 ec2-user apache 44 May 11 03:02 . drwxrwsr-x 2 ec2-user apache 6 May 11 03:02 inc drwxr-xr-x 20 root root 280 May 11 02:07 .. drwxrwsr-x 2 ec2-user apache 6 Apr 12 12:01 cgi-bin drwxrwsr-x 2 ec2-user apache 6 Apr 12 12:01 html [ec2-user@ip-10-22-0-190 www]\$ sudo yum install -y httpd Loaded plugins: extras_suggestions, langpacks, priorities, update-motd Package httpd-2.4.53-1.amzn2.x86_64 already installed and latest version Nothing to do AWS Secret Access Key [None]: [REDACTED] Default region name [None]: us-east-1 Default output format [None]: json [ec2-user@ip-10-22-0-190 www]\$ aws s3 cp s3://cf2-s3-bucket/dbinfo.inc /var/www/inc/ download: s3://cf2-s3-bucket/dbinfo.inc to inc/dbinfo.inc [ec2-user@ip-10-22-0-190 www]\$ aws s3 cp s3://cf2-s3-bucket/SamplePage.php /var/www/html/ download: s3://cf2-s3-bucket/SamplePage.php to html/SamplePage.php [ec2-user@ip-10-22-0-190 www]\$</pre>			

RDS MySQL setups

- Connect to the RDS using the endpoint
- Show the starting databases
- Create a table by defining three columns: ID, Name, and Address.
- Insert a few records and verify by querying for all field values.

```
[ec2-user@ip-10-22-0-190 www]$ mysql -h cfn-rds-inst01.cyzpvqfcc7eq.us-east-1.rds.amazonaws.com -P 3306 -u dbuser -p
```

```
MySQL [(none)]>show databases;
```

```
+-----+  
| Database      |  
+-----+  
| information_schema |  
| mysql          |  
| performance_schema || sys      |  
| wpdb01         |  
+-----+  
5 rows in set (0.00 sec)
```

```
MySQL [(none)]> use wpdb01;
```

```
Database changed
```

```
MySQL [wpdb01]> CREATE TABLE EMPLOYEES(ID int(11) UNSIGNED AUTO_INCREMENT,  
-> NAME VARCHAR(45),  
-> ADDRESS VARCHAR(90),  
-> PRIMARY KEY (ID));
```

```
Query OK, 0 rows affected, 1 warning (0.03 sec)
```

```
MySQL [wpdb01]> INSERT INTO EMPLOYEES(NAME, ADDRESS)
```

```
-> VALUES ('John Frank', '7811 Price Rd, VA 23334'),  
-> ('Jason M. Alex', '44 Fort Rd, VA 22334'),  
-> ('Julia G. JJ', '45 Wfg Rd, VA 23334');
```

```
Query OK, 3 rows affected (0.01 sec)
```

```
Records: 3  Duplicates: 0  Warnings: 0
```

```
MySQL [wpdb01]> select * from wpdb01.EMPLOYEES;
```

```
+-----+  
| ID | NAME      | ADDRESS      |  
+-----+  
| 1 | John Frank | 7811 Price Rd, VA 23334 |  
| 2 | Jason M. Alex | 44 Fort Rd, VA 22334 |  
| 3 | Julia G. JJ | 45 Wfg Rd, VA 23334 |  
+-----+  
3 rows in set (0.00 sec)
```

```
MySQL [wpdb01]>
```

Now let's verify if the webserver is running.

Using IPv4 Public address (3.87.155.27) of the EC2 instance on the web browser shows the Apache default "Test Page" as expected.

Or using Public DNS (IPv4) <https://ec2-3-87-155-27.compute-1.amazonaws.com/> produce the same result. That shows we are good to go to the next step.



If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

If you are the website administrator:

You may now add content to the directory /var/www/html/. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file /etc/httpd/conf.d/welcome.conf.

You are free to use the image below on web sites powered by the Apache HTTP Server:



When appending "SamplePage.php" on IPv4 ip as: "3.87.155.27/SamplePage.php" the expected Sample Page showed correctly. Starting page records were showing from the data entered from AWS CLI.

A screenshot of a web browser window. The address bar shows 'Not secure | 3.87.155.27/SamplePage.php'. The main content area has a large heading 'Sample page'. Below the heading, there is a table with columns 'NAME' and 'ADDRESS'. A button 'Add Data' is located to the right of the ADDRESS column. Below the table, there is another table showing data records. The first table has columns 'NAME' and 'ADDRESS'. The second table has columns 'ID', 'NAME', and 'ADDRESS'. The data in both tables is identical.

Now let's add a few more data records using the Sample page. The last two records were added as expected.

A screenshot of a web browser window. The address bar shows 'Not secure | 3.87.155.27/SamplePage.php'. The main content area has a large heading 'Sample page'. Below the heading, there is a table with columns 'NAME' and 'ADDRESS'. A button 'Add Data' is located to the right of the ADDRESS column. Below the table, there is another table showing data records. The first table has columns 'NAME' and 'ADDRESS'. The second table has columns 'ID', 'NAME', and 'ADDRESS'. The data in both tables is identical, including the newly added records.

Set up a scaled and load-balanced for the webserver application

Create an AMI image from ec2 instance that was created from the initial template.

Amazon Machine Images (AMIs) (1/1) [Info](#)

Name	AMI ID	AMI name	Source	Owner	Visibi...	Status	Creation date
-	ami-065050ef51452cca7	cfn-ami-lynux	046672410778/cfn-ami-lynux	046672410778	Private	Available	2022/05/11 15:27 GMT-4

AMI ID: ami-065050ef51452cca7

Details | Permissions | Storage | Tags

AMI ID	Image type	Platform details	Root device type
ami-065050ef51452cca7	machine	Linux/UNIX	EBS
AMI name	Owner account ID	Architecture	Usage operation
cfn-ami-lynux	046672410778	x86_64	RunInstances
Root device name	Status	Source	Virtualization type
/dev/xvda	Available	046672410778/cfn-ami-lynux	hvm
Boot mode	State reason	Creation date	Kernel ID
-	-	Wed May 11 2022 15:27:29 GMT-0400 (Eastern Daylight Time)	-
Block devices	Description	Product codes	RAM disk ID
/dev/xvda=snap-025ceb9a624e1b16c8:true:gp2	cfn-ami-lynux	-	-
Deprecation time	Last launched time		

Create a launch template

EC2 > Launch templates

Launch templates (1/1) [Info](#)

Launch template ID	Launch template name	Default version	Latest version	Create time	Created by
lt-04af2c9e9fa9df51f	cfn-ec2-launch-template	1	1	2022-05-11T19:20:45.000Z	arn:aws:iam::046672410778:root

Details | Versions | Template tags

Launch template version details

Version	Description	Date created	Created by
1 (Default)	1	2022-05-11T19:20:45.000Z	arn:aws:iam::046672410778:root

Actions | Delete template versic

Instance details | Storage | Resource tags | Network interfaces | Advanced details

AMI ID	Instance type	Availability Zone	Key pair name
ami-0022f774911c1d690	t2.micro	-	abebe_key1
Security groups	Security group IDs		
-	sg-0b6e4d5dc1fc841		

Create a launch configuration

EC2 > Launch configurations

Launch configurations (1/1) Info					
<input type="button" value="C"/> Actions ▾ Copy to launch template ▾					
<input checked="" type="checkbox"/>	Name	AMI ID	Instance type	Spot price	Creation time
<input checked="" type="checkbox"/>	cfn-ec2-launch-configuration	ami-065050ef51452cca7	a1.medium	-	Wed May 11 2022 15:29:42 GMT-0400 (Eastern Daylight Time)

Launch configuration: cfn-ec2-launch-configuration

Details		
AMI ID ami-065050ef51452cca7	Instance type a1.medium	IAM instance profile arn:aws:iam::046672410778:instance-profile/cfn-ec2-launch-configuration
Kernel ID -	Key name abebbe_key1	Monitoring true
EBS optimized false	Security groups sg-0b6e4d5dc1fc841	Spot price -
Create time Wed May 11 2022 15:29:42 GMT-0400 (Eastern Daylight Time)	RAM disk ID -	IP address type Default
Metadata accessible -	Token hop limit -	Metadata version -

Created a target group

EC2 > Target groups

Target groups (1/1) Info						
<input type="button" value="C"/> Actions ▾						
<input checked="" type="checkbox"/>	Name	ARN	Port	Protocol	Target type	Load balancer
<input checked="" type="checkbox"/>	cfn-tg	arn:aws:elasticloadbalancing:us-east-1:046672410778:targetgroup/cfn-tg/00b091066e5af6d2	80	HTTP	Instance	cfn-alb

cfn-tg

[arn:aws:elasticloadbalancing:us-east-1:046672410778:targetgroup/cfn-tg/00b091066e5af6d2](#)

Details	Targets	Monitoring	Health checks	Attributes	Tags
Details					
Target type Instance	Protocol : Port HTTP: 80	Protocol version HTTP1	VPC vpc-0b70ffff53a92cea68		
IP address type IPv4	Load balancer cfn-alb				
Total targets 3	Healthy 0	Unhealthy 2	Unused 0	Initial 0	

Created Application Load Balancer

[Create Load Balancer](#) [Actions](#)

Filter by tags and attributes or search by keyword

Name	DNS name	State	VPC ID	Availability Zones	Type	Created At
at-alb	at-alb-943798799.us-east-1.elb.amazonaws.com	Active	vpc-04a740dffaefc0e74	us-east-1a, us-east-1b	application	May 2, 2022 at 10:48:56 PM
cfn-alb	cfn-alb-1220193060.us-east-1.elb.amazonaws.com	Active	vpc-0b70ff53a92cea68	us-east-1b, us-east-1a	application	May 11, 2022 at 3:45:59 PM

Load balancer: cfn-alb

Description [Listeners](#) [Monitoring](#) [Integrated services](#) [Tags](#)

Basic Configuration

Name: cfn-alb
ARN: arn:aws:elasticloadbalancing:us-east-1:046672410778:loadbalancer/app/cfn-alb/194d2eaa4cf081d7
DNS name: cfn-alb-1220193060.us-east-1.elb.amazonaws.com (A Record)
State: Active
Type: application
Scheme: internet-facing
IP address type: ipv4 [Edit IP address type](#)
VPC: vpc-0b70ff53a92cea68
Availability Zones: subnet-0b1eaddd1a0a1f41f - us-east-1b (IPv4 address: Assigned by AWS)
subnet-0e846c4e48b6cf73a - us-east-1a (IPv4 address: Assigned by AWS)

Create an Auto Scaling group

[EC2](#) > [Auto Scaling groups](#)

Auto Scaling groups (1/1)

[Create an Auto Scaling group](#)

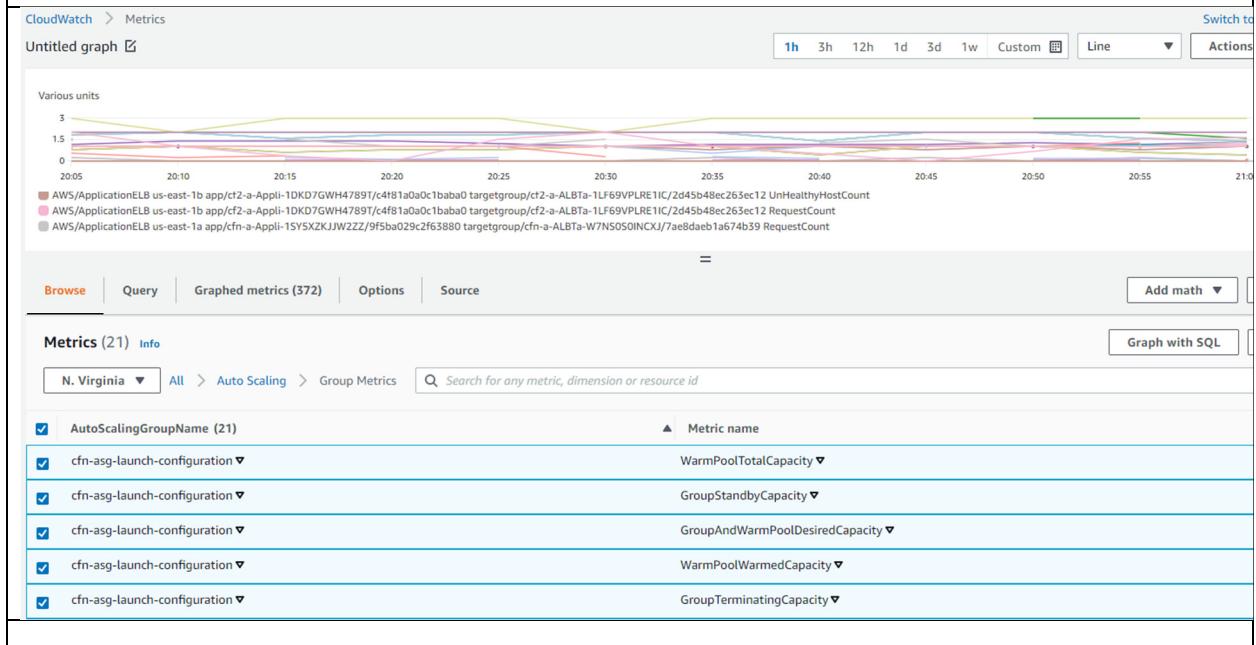
Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zone
cfn-asg-launch-configuration	cfn-ec2-launch-template Version Default	1	-	1	1	1	us-east-1a, us-east-1b

[Details](#) [Activity](#) [Automatic scaling](#) [Instance management](#) [Monitoring](#) [Instance refresh](#)

Instances (1)

Instance ID	Lifecycle	Instance type	Weighted capacity	Launch template/configuration	Availability Zone	Health status	Protected from
i-0905d53634e4acb4e	InService	t2.micro	-	cfn-ec2-launch-template Version 1	us-east-1b	Healthy	

Checked on CloudWatch Metrics.



Clean Up steps taken After completing the project tasks.

1. delete your Auto Scaling group
2. delete your launch template
3. delete your launch configuration
4. delete your load balancer
5. delete your target group
6. **delete the stacks created**