

1. Deploy a EC2 instance using of t2. micro instance in Mumbai Region in zone ap-south-1a. Configure web server on it and make it live. and attach a 5gb ebs volume. where create 10 files. like training.txt
2. My one team is working with In Mumbai India Region and they have some important data in the ebs volume. And another team is working with Singapore region. They also want to access same data which is available in Mumbai Region. Share it in different region.
3. I have on web server in Mumbai Region where my website is running. I need same server in Singapore region. Migrate this web server from Mumbai to Singapore.
4. Launch a AWS S3 bucket with Uniq name. and upload some objects. And this S3 bucket should be reachable on windows host from where I can upload the object in the AWS S3 bucket.
5. Create a custom VPC. Where you need to create two subnets like Private subnet and Public subnet. in the public subnet I want to host my web server. Where my website is running and Private subnet my database is running. Data base should not be reachable publically.
6. Create two custom VPC one in Mumbai Region and Another is in Singapore Region. so configure VPC peering in between Mumbai and Singapore

7. Deploy an EC2 instance using with Cloud Formation in Mumbai Region ap-south-1a zone. Instance should be reachable.
8. We people are working on a common project in a sale region. But my servers are in different zones. So I want to share project information with everyone simultaneously. Configure EFS storage it should be mount on every server.
9. Enable MFA on root account on AWS account, and generate access key and secret key for root account.
10. Create an IAM role for Cloud Formation service with administrator full access. And create a stack to deploy of your JSON code

File Edit View

Question 7

use AWS notepad file

I

Ln 4, Col 21 33 characters

100% Windows (CRL)





ChatGPT

Certainly! Here's a step-by-step procedure to deploy an EC2 instance using AWS CloudFormation in the Mumbai region (ap-south-1) and ensure it's reachable:

Step 1: Prepare your CloudFormation Template

Create a CloudFormation template (YAML or JSON format) that defines your EC2 instance. Here's a basic example template:

yaml

Copy code

```
AWSTemplateFormatVersion: '2010-09-09'  
Resources:  
  MyEC2Instance:  
    Type: AWS::EC2::Instance  
    Properties:  
      AvailabilityZone: ap-south-1a  
      ImageId: ami-0c1a7f89451184c8b # Amazon Linux 2 AMI ID  
      InstanceType: t2.micro  
      KeyName: your-key-pair-name  
      SecurityGroups:  
        - your-security-group-name
```

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Search



ENG
IN



03:56 PM
14-04-2024

DELL

Replace 'ami-0c1a7f89451184c8b' with your desired AMI ID, 'your-key-pair-name' with your key pair name, and 'your-security-group-name' with your security group name.

Step 2: Deploy the CloudFormation Stack

1. Go to the AWS Management Console and navigate to the CloudFormation service.
2. Click on "Create stack" > "With new resources (standard)".
3. Choose "Upload a template file" and upload your prepared CloudFormation template.
4. Click "Next".
5. Enter a stack name (e.g., MyEC2Stack) and provide any necessary parameters.
6. Click "Next" through the options until you reach the Review page.
7. Review your stack configuration and click "Create stack".

Step 3: Monitor Stack Creation Progress

Once you've created the stack, CloudFormation will start provisioning the resources defined in your template. You can monitor the progress in the CloudFormation console.

Step 4: Access EC2 Instance

1. Once the stack creation is complete, navigate to the EC2 service in the AWS Management Console.

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.



Search



ENG
IN



03:56 PM
14-04-2024

DELL

ChatGPT 3.5 ▾

Step 4: Access EC2 Instance

1. Once the stack creation is complete, navigate to the EC2 service in the AWS Management Console.
2. Find the newly created instance and note down its public IP or DNS.
3. Try accessing the instance via SSH (if applicable) or HTTP (if you've configured a web server as in the UserData script).

Step 5: Verify Reachability

Attempt to connect to your EC2 instance using the public IP or DNS. If successful, you've ensured that your instance is reachable.

Step 6: Cleanup (Optional)

Remember to delete the CloudFormation stack once you're done with your testing to avoid incurring unnecessary costs.

That's it! You've successfully deployed an EC2 instance using AWS CloudFormation in the Mumbai region and ensured its reachability.



Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.



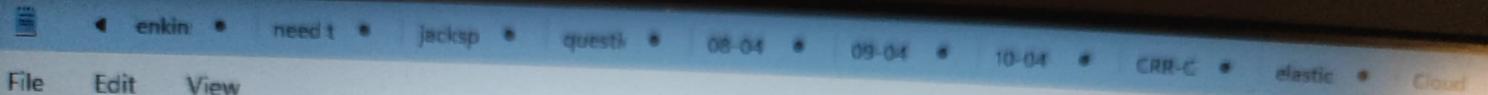
Search



ENG
IN



03:51
14-04



Question 10

use IAM notepad file

**ChatGPT**

Certainly! Below are the step-by-step instructions to create an IAM role for CloudFormation service with administrator access and then deploy a stack using your JSON CloudFormation template:

Step 1: Create an IAM Role for CloudFormation

1. Go to the AWS Management Console and navigate to the IAM service.
2. In the left-hand menu, click on "Roles", then click "Create role".
3. Choose "AWS service" as the type of trusted entity.
4. Select "CloudFormation" from the list of services.
5. Click "Next: Permissions".
6. In the "Attach permissions policies" step:
 - Search for "AdministratorAccess" in the search box.
 - Check the box next to "AdministratorAccess" to attach the policy.
7. Click "Next: Tags" (Optional) and add any tags if necessary.
8. Click "Next: Review".
9. Provide a name for your role (e.g., CloudFormationAdminRole) and optionally add a description.
10. Click "Create role".



Step 2: Prepare your CloudFormation Template

Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.



Search



ENG
IN



04:22 PM
14-04-2024

DELL

Step 2: Prepare your CloudFormation Template

Ensure you have your CloudFormation JSON template ready. This should include your desired resources, such as an EC2 instance, IAM roles, etc. Here's a basic example:

json

Copy code

```
"AWSTemplateFormatVersion": "2010-09-09",
"Resources": {
    "MyEC2Instance": {
        "Type": "AWS::EC2::Instance",
        "Properties": {
            "ImageId": "ami-0c1a7f89451184c8b",
            "InstanceType": "t2.micro",
            "KeyName": "your-key-pair-name",
            "SecurityGroups": ["your-security-group-name"],
            "UserData": {
                "Fn::Base64": [
                    "Fn::Join": [
                        "",
                        [
                            "#!/bin/bash\n"

```

Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.



Search



PT3.5

```
SecurityGroups -> ["your-security-group-name"],  
"UserData": {  
    "Fn::Base64": {  
        "Fn::Join": [  
            "",  
            [  
                "#!/bin/bash\n",  
                "echo \"Hello, World!\" > /var/www/html/index.html\n",  
                "yum update -y\n",  
                "yum install -y httpd\n",  
                "systemctl start httpd\n",  
                "systemctl enable httpd\n"  
            ]  
        ]  
    }  
}
```

Replace `{"ami-0c1e7f89451184c8b"}` , `{"key-pair-name"}` , and `{"your-security-group-name"}` with appropriate values.

Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.



Search



ENG
IN



0

Replace `{"ami-0c1a7f89451184e8b"`, `{"youz-key-pair-name"}`, and `{"youz-security-group-name"}` with appropriate values.

Step 3: Deploy the CloudFormation Stack

1. Go to the AWS Management Console and navigate to the CloudFormation service.
2. Click on "Create stack".
3. Choose "Template is ready" and "Upload a template file".
4. Upload your prepared CloudFormation JSON template.
5. Click "Next".
6. Provide a stack name (e.g., MyEC2Stack) and optionally add any parameters or tags.
7. Click "Next".
8. Review your stack configuration and click "Create stack".
9. Wait for the stack creation to complete. You can monitor progress in the CloudFormation console.

Step 4: Verify the Stack and Resources

Once the stack creation is complete, navigate to the CloudFormation console and select your stack to view details. Ensure that all resources were created successfully without any errors.

Step 5: Access EC2 Instance (If Applicable)

Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.



Search



ENG
IN



04:23 PM
14-04-2024

DELL

com/c94a675a3-1c08-4ccb-92a6-7cd7c66d7ae8

ps IPLT20.com - Indian... Typing practice Data Structure and... V S B ENGINEERING... INET COMMUNICAT... LTI



ChatGPT 3.5 ▾

4. Upload your prepared CloudFormation JSON template.
5. Click "Next".
6. Provide a stack name (e.g., MyEC2Stack) and optionally add any parameters or tags.
7. Click "Next".
8. Review your stack configuration and click "Create stack".
9. Wait for the stack creation to complete. You can monitor progress in the CloudFormation console.

Step 4: Verify the Stack and Resources

Once the stack creation is complete, navigate to the CloudFormation console and select your stack to view details. Ensure that all resources were created successfully without any errors.

Step 5: Access EC2 Instance (If Applicable)

If your CloudFormation template creates an EC2 instance, you can access it using the public IP or DNS provided in the EC2 console.

That's it! You've successfully created an IAM role for CloudFormation with administrator access and deployed a stack using your JSON CloudFormation template.



Message ChatGPT...



?

ChatGPT can make mistakes. Consider checking important information.



Search



ENG
IN



04:23 PM
14-04-2024