Operations Playbook

Name: Spurthy Mutturaj

Date: 17 Sep, 2023

Class/Semester: IFT 562, 4th Semester

Contents

How to connect to the Mom & Pop Cafe Test EC2 instance

NOTE:I am using a MAC

1. Ensure you have a copy of pem file used to authenticate with your instance.
2. Open terminal and change permission on the pem file to read only.
3. Run the ssh command on the terminal to connect to the EC2 instance passing the following parameters and options
    - ssh -i private_key.pem ec2-user@public_ip-address

How to use the AWS CLI to connect to your AWS account

1. run the aws configure command
2. Each instance will have secret and access keys created at the set up of instance.
3. When promted by the cli, enter the following details:
    - AWS Access Key ID
    - AWS Scret Access Key ID
    - Default region: us-east-1
    - Default O/P format: json

How to make a modification to the lab policy using the AWS CLI

1. Using the aws cli:
    a. List the policies using the command aws iam list-policies –scope local
    b. Using get-policy and the arn value from the previous step and the version-id pipe it to a json file:
    get-policy --policy-arn <arn value> --version-id <version id> > file_name.json
    get the lab_policy
2. Open the downloaded lab_policy in json format using a vi editor.
3. Make the necessary changes
4. Use the aws iam create-policy-version command to create a new version of the managed policy. Replace <value> with your own values.
    a. aws iam create-policy-version \

    ```
      --policy-arn <ar value> \

     --policy-document <file:// the edited policy path file_name.json>

    -- set-as-default
    ```

In the last step the option - set as default, uses the new edited json file as the iam policy.

After running the command, you will receive a JSON response indicating the success of the operation. Make sure to review it for any errors or issues.

How to add a parameter to the parameter store for allowing cookies on the website

1. Sign in to AWS Management Console:
    a. Open your web browser and navigate to the AWS Management Console.
    b. Sign in to your AWS account using your credentials.

2. Access Systems Manager Parameter Store:
    a. In the AWS Management Console, navigate to the "Services" menu.
    b. Under the "Management & Governance" section, select "Systems Manager."
3. Create a New Parameter:
    a. In the Systems Manager dashboard, click on "Parameter Store" in the left navigation pane.
    b. Click the "Create Parameter" button.
4. Configure Parameter:
    a. Fill in the parameter details:
        i. Name: /web.config/cookie_toggle.
        ii. Description: This feature allows you to turn cookies on or off for the Cafe website.
        iii. Value: "True"
5. We Enter Parameter Value:
    a. In the "Parameter value" section, the value for allowing cookies, is True.
    b. Optionally, you can leave the other options as default.
6. Create Parameter:
    a. Click the "Create parameter" button to create the parameter.

How to connect to an EC2 instance to describe instances

1. Open your terminal or command prompt.
2. Use the ssh command to connect to your EC2 instance. Replace <your-instance-IP> with your EC2 instance's public IP address, and <your-key.pem> with the path to your private key file.
    a. ssh -i <your-key.pem> ec2-user@<your-instance-IP>
3. Once connected to the EC2 instance, you can use the AWS CLI to describe instances. Run the following command to list all EC2 instances in your account:
    a. aws ec2 describe-instances
4. You will receive a JSON response containing information about your EC2 instances.
5. To exit the SSH connection, simply type:
    a. exit

These steps will allow you to connect to your EC2 instance and use the AWS CLI to describe instances

How to launch an EC2 instance

1. To launch an instance with specified details, first make a ssh conncetion to the ec2-instance using a key-pair .pem file on mac OS.
2. Use aws cli and specify the details as follows:

```
aws ec2 run-instances \

  --image-id $AMI \   # Replace with the actual Amazon Linux 2 AMI ID

  --instance-type t1.micro \

  --key-name YourKeyName \          # Replace with your key pair name
```

```
        --security-group-ids sg-xyz\  # Replace with your security group ID

        --subnet-id subnet-xyz       # Replace with your subnet ID
```

Note. : to get the Amazon Linux2 AMI ID, use the systems parameter store:

```
AMI=$(aws ssm get-parameters --names /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2 --query 'Parameters[0].[Value]' --output text)
```

The above steps will launch an EC2 Instance in the cloud

How to fix a misconfigured web server with (_________) issue

1. The issues is security-group's inbound rules.
2. The webserver security group doesn't allow ssh traffic , and hence when we try to connect using the terminal, it does not allow the connection.
3. To fix it , In the AWS UI-> s(ervices menu)EC2 ->instances, select the Misconfigured Web Server and in the security group tab , open the security groups and click on edit the inbound rules:
    a. Add a rule
    b. Set type-ssh
    c. Source type-anywhere-IPv4
    d. Click save rules
4. Make an ssh connection using the private_key.pem :
    a. Ssh -i private_key.pem ec2-user@<public ip address of Misconfigured Web Server >

How to change the AMI instance on the create-lamp-instance.sh script

1. Establish SSH Connection: Begin by establishing an SSH connection with an EC2 instance. This initial step is crucial as it sets the stage for making changes to the AMI instance.
2. Configure AWS Environment: In the second stage, ensure your AWS environment is configured to meet the specific requirements of your task. This includes ensuring you have the necessary permissions and access to EC2 instances.
3. Backup and Open Script: Before making any changes, create a backup of the create-lamp-instance.sh script. Then, open this script in a text editor. Inside the script, take note of essential information such as VPC settings, Subnet ID, Security Group, and Instance ID.
4. Troubleshoot Errors: If you encounter any errors, especially if you see an error message like "InvalidAMIID.nooound," carefully review the script. The error message provides a clue about the line where the issue originates. Identify and address any technical errors.
5. Update AMI ID: Within the script, you will find an existing AMI ID. Replace this AMI ID with the exact AMI ID of the instance you intend to use. This step is critical to ensure that you are specifying the correct AMI instance.
6. Rerun the Script: After making all the necessary adjustments and confirming that you have entered the correct AMI ID, rerun the create-lamp-instance.sh script. This action ensures that the AMI ID has been accurately updated and modified as intended.

How to tail a log in Linux

1. To tail a log, example-file.log located in a filepath:

    Use command: sudo tail -f filepath/ example-file.log

The above will display the live output of the cloud-init process (example log below):

```
Cloud-init v. 21.2-3.el8  running 'modules:config' at Fri, 10 Sep 2023 15:24:11 +0000. Up 52.68 seconds.

Cloud-init v. 21.2-3.el8  running 'modules:final' at Fri, 10 Sep 2023 15:24:12 +0000. Up 53.44 seconds.

Cloud-init v. 21.2-3.el8  finished at Fri, 10 Sep 2023 15:24:12 +0000. Datasource DataSourceEc2.  Up 53.58 seconds
```

2. Using the option -n , the command specifies the number of lines from the end of a file that should be displayed.
   ex. tail -n 10 example-file.log
   This will show you the last 10 lines of the log file.
3. Cntl+c is - sends an interrupt signal to the currently running process. This is typically used to stop or terminate the running process.
4. If you want to stop, type ”:q”

How to create an Auto Scaling Group in the AWS UI

1. Access the AWS Management Console:
   a. Open a web browser and navigate to the AWS Management Console (https://aws.amazon.com/).
   b. Sign in with your AWS account credentials.
2. Navigate to Auto Scaling Groups:
   a. From the AWS Management Console, click on "Services" in the top-left corner.
   b. Under the "Compute" section, select "Auto Scaling."
3. Create Auto Scaling Group:
   a. In the Auto Scaling dashboard, scroll down and click on "Create Auto Scaling group."
4. Configure Launch Template and Network:
   a. In the configuration step, provide the necessary details:
      i. Auto Scaling group name: Enter a unique name.
      ii. Choose the appropriate launch template or configuration.
      iii. Click "Next" to proceed.
   b. Set up network configurations:
      i. Select the desired VPC.
      ii. Choose the relevant subnets.
      iii. Click "Next" to continue.
5. Load Balancing and Additional Settings:
   a. Configure load balancing:
      i. Select "Attach to an existing load balancer."
      ii. Choose the target group for load balancing.
   b. In the "Additional settings" section:
      i. Enable group metrics collection within CloudWatch.

        ii. Click "Next."

6. Configure Group Size and Scaling Policies:
    a. Set the group size:
        i. Desired capacity: Enter 5 (Minimum: 5 nodes).
        ii. Minimum capacity: Enter 5 (Minimum: 5 nodes).
        iii. Maximum capacity: Enter 10 (Maximum: 10 nodes).
    b. Configure scaling policies:
        i. Select "Target tracking scaling policy."
        ii. Scaling policy name: Enter a name (e.g., MyScalingPolicy).
        iii. Metric type: Choose "Average CPU utilization."
        iv. Target value: Enter 50 (Target: 7 nodes).
7. Add Tags:
    a. On the "Add tags" page, click "Add tag" and configure:
        i. Key: Enter a tag key (e.g., Name).
        ii. Value: Enter a tag value (e.g., WebApp).
8. Review and Create:
    a. Review the Auto Scaling group configuration details.
    b. Once satisfied, click the "Create Auto Scaling group" button to create the group.

Your Auto Scaling group with step scaling, targeting a minimum of 5 nodes, a maximum of 10 nodes, and a target of 7 nodes, is now created and configured. It will automatically adjust the number of instances based on the defined CPU utilization target, ensuring efficient resource management.

To create a step scaling policy for scale out (increase capacity):

1. Open the Amazon EC2 console and navigate to "Auto Scaling Groups" from the navigation pane.
2. Select the checkbox next to your Auto Scaling group.
3. Verify and adjust the minimum and maximum size limits for your group if needed.
4. Go to the "Automatic scaling" tab and choose "Create dynamic scaling policy."
5. For the policy type, select "Step scaling."
6. Specify a name for the policy.
7. Choose an existing CloudWatch alarm or create a new one to monitor a metric like CPU utilization (set the alarm threshold to >= 80%).
8. Define the change in group size for this policy when executed (e.g., Add 30% of the group size).
9. Optionally, add more steps with specific scaling adjustments.
10. Choose "Create."

To create a step scaling policy for scale in (decrease capacity):

1. Continue from where you left off after creating the scale-out policy.
2. For the policy type, choose "Step scaling."
3. Specify a name for the policy.
4. Choose an existing CloudWatch alarm or create a new one to monitor the metric (e.g., CPU utilization <= 40%).
5. Define the change in group size for this policy when executed (e.g., Remove 2 capacity units).

6. Optionally, add more steps for scaling in.
7. Choose "Create."

How to create a Route 53 health check

1. Access Route 53:
    - a. Access the AWS Management Console.
    - b. From the Services menu, choose Route 53.
2. Create Health Check:
    - a. In the left navigation pane, click Health checks.
    - b. Click Create health check.
3. Configure Health Check:

    - a. Configure the following settings:
        - i. Name: Provide a name for the health check.
        - ii. What to monitor: Choose an appropriate option (e.g., Endpoint).
        - iii. Specify endpoint by: Select the method (e.g., IP address).
        - iv. IP address: Enter the target IP address.
        - v. Path: Specify the path to check (optional).
    - b. Expand Advanced configuration:
        - i. Configure settings like request interval and failure threshold according to your needs.
    - c. Click Next.
4. Create Alarm and Notification:
    - a. Configure the following:
        - i. Create alarm: Choose Yes.
        - ii. Send notification to: Select or create an SNS topic.
        - iii. Topic name: Provide a name for the SNS topic.
        - iv. Recipient email address: Enter an email address for notifications.
    - b. Click Create health check.

The Route 53 health check will now periodically monitor the specified endpoint and send notifications based on your configured settings. You can check the health check's status and monitoring information in the Route 53 console, and you'll receive email notifications if any issues are detected.

How to create an Amazon RDS instance using the CLI

1. Configure you AWS CLI
2. An example command to create an RDS instance:
    - aws rds create-db-instance \
    - --db-instance-identifier YourDBInstanceName \
    - --db-instance-class db.t2.micro \
    - --engine mysql \
    - --allocated-storage 20 \
    - --master-username YourDBUsername \

- --master-user-password YourDBPassword \
- --vpc-security-group-ids YourSecurityGroupID \
- --availability-zone YourAvailabilityZone \
- --db-subnet-group-name YourSubnetGroupName

  [Modify the parameters as needed for your specific instance]

3. After running the create-db-instance command, AWS will initiate the creation of your RDS instance. You can monitor the status using this command
    - aws rds describe-db-instances --db-instance-identifier YourDBInstanceName
4. Once the RDS instance is available, you can access it using the endpoint provided. You can retrieve the endpoint with the following command:
    - aws rds describe-db-instances \
    - --db-instance-identifier YourDBInstanceName \
    - --query "DBInstances[0].Endpoint.Address" \
    - --output text

5. Connect and Use Your RDS Instance: Now that you have created your RDS instance, you can connect to it using a MySQL client or any compatible database tool and start using it for your applications. Be sure to use the master username and password you specified during instance creation.

How to collect information about an instance

1. To collect information about an instance in AWS using the AWS CLI, you can use the following commands

    a. This command will return the instance ID, instance type, public DNS name, public IP address, availability zone, VPC ID, and security group IDs for the specified instance.
    # Replace "YourInstanceName" with the name of your instance.

    - aws ec2 describe-instances \
    - --filters "Name=tag:Name,Values=YourInstanceName" \
    - --query "Reservations[*].Instances[*].[InstanceId,InstanceType,PublicDnsName,PublicIpAddress,Placement.AvailabilityZone,VpcId,SecurityGroups[*].GroupId]"

    b. This command will return the IPv4 CIDR block of the VPC associated with the instance.

    - *# Replace "YourVPCID" with the VPC ID obtained in Step 1.*
    - aws ec2 describe-vpcs --vpc-ids YourVPCID \
    - --query "Vpcs[*].CidrBlock"

    c. This command will return the subnet ID and IPv4 CIDR block of the subnet(s) associated with the instance's VPC.

    a. *# Replace "YourVPCID" with the VPC ID obtained in Step 1.*
    b. aws ec2 describe-subnets \
    c. --filters "Name=vpc-id,Values=YourVPCID" \

d. --query "Subnets[*].[SubnetId,CidrBlock]"

d. This command will return a list of Availability Zones in the specified region.

- *# Replace "<region>" with your AWS region (e.g., us-east-1 or eu-west-2).*
- aws ec2 describe-availability-zones \
- --filters "Name=region-name,Values=<region>" \
- --query "AvailabilityZones[*].ZoneName"

Note: Make sure to replace the placeholders ("<region>", "YourInstanceName", "YourVPCID") with the actual values and run these commands in your AWS CLI to collect the instance information.

How to create two subnets in a subnet group via the AWS CLI

1. Create Private Subnet 1
   - aws ec2 create-subnet \
   - --vpc-id <YourVPCID> \
   - --cidr-block <CIDR block for Private Subnet 1> \
   - --availability-zone <Availability Zone for Private Subnet 1>

2. Create Private Subnet 2
   - aws ec2 create-subnet \
   - --vpc-id <YourVPCID> \
   - --cidr-block <CIDR block for Private Subnet 2> \
   - --availability-zone <Availability Zone for Private Subnet 2>

3. Create RDS Subnet Group
   - aws rds create-db-subnet-group \
   - --db-subnet-group-name MyDBSubnetGroup \
   - --db-subnet-group-description "Subnet group for my RDS instance" \
   - --subnet-ids <SubnetId for Private Subnet 1> <SubnetId for Private Subnet 2>

How to use the mysqldump tool to take a backup of a SQL database and restore it on another SQL instance

1. Backup the SQL Database:
   - Open an SSH session to the source SQL instance.
   - Use the `mysqldump` utility to create a backup of the source database.
   - mysqldump --user=<username> --password='<password>' --databases <database_name> > backup.sql
2. Open an SSH session to the target SQL instance.
   Use the `mysql` command to restore the backup to the target database.

```
mysql --user=<username> --password='<password>' --host=<target_host> < backup.sql
```

Replace `<username>`, `<password>`, and `<database_name>` with your MySQL credentials and the database you want to backup.

3. Verify that the database on the target instance was successfully created and populated by running SQL queries or examining the data as needed.

How to enable VPC Flow Logs via the command line interface

1.

How to troubleshoot network connectivity on an instance

1.

How to take a snapshot of an EBS volume

1.

How to synchronize files using the command line (aws s3api and aws s3)

1.

How to create a S3 bucket via the CLI

1.

How to add an event notification to a S3 bucket

1.

How to install the CloudWatch Agent

1.

How to create a CloudWatch Events/CloudWatch EventBridge notification rule

1.

How to use the prebuilt stopinator script to turn off instances with the tag value of your full name

1.

How to resize an EC2 instance using the AWS CLI

1.

How to detect drift in a CloudFormation template

1.

How to create an Amazon Athena table

1.

How to manually review access logs to find anomalous user activity

1.

How to create a batch file to update the café website to change its colors

1. Create an empty file - filename.sh , using the touch command and open it in a vi editor.

2. Write a script with the following commands:
    - #!/bin/bash   - she-bang
      aws s3 cp ~/sysops-activity-files/ s3://<my-bucket>/ --recursive --acl public-read
      Note: replace <my-bucket> with the name of your bucket.
3. Make the file executable : chmod +x filename.sh
4. Open the index file of the website in a text editor
5. Locate and modify the color attributes , ex set bgcolor=Cyan , bgcolor=Coral, bgcolor=Brown. And save the file
6. Run the batch file using the command: ./filename.sh

How to create a Lambda Layer and add it to a Lambda function

1. Create a Lambda Layer
    a. Go to the AWS Management Console and select Services > Lambda.
    b. Choose Layers.
    c. Click Create layer.
    d. Configure the layer settings:
    e. Name: [LayerName]
    f. Description: [LayerDescription]
    g. Code entry type: Upload a .zip file
    h. Choose Upload and select the [LayerZipFileName].zip file you downloaded.
    i. Compatible runtimes: Python 3.7 (or your preferred runtime).
    j. Click Create.
2. Create the Lambda Function
    Configure the function as follows:
    a. Author from scratch: Selected
    b. Function name: [FunctionName]
    c. Runtime: Python 3.7 (or your preferred runtime)
    d. Expand Change default execution role and select "Use an existing role."
    e. Choose the existing role: [ExistingRoleName]
    f.
    g. Click Create function.

3. Add the Lambda Layer to the Function
    a. In the Function overview panel, under [FunctionName], choose Layers.
    b. Click Add a layer.
    c. In the Add layer page, configure as follows:
    d. Choose a layer: Select the Custom layers card.
    e. Custom layers: [LayerName]
    f. Version: [LayerVersion]
    g. Click Add.

How to create a Lambda function from a prebuilt package

1. Access AWS Lambda Console: Log in to your AWS Management Console and navigate to the AWS Lambda service.
2. Create a New Lambda Function:

    a. Click on "Create function."
    b. Choose "Author from scratch."
    c. Fill in the function name, runtime (e.g., Python, Node.js), and existing role or create a new one.
3. Configure Function
    a. In the "Function code" section, select "Upload a .zip file."
    b. Upload a prebuilt deployment package that contains your function code.
    c. Define the handler (entry point) for your function.
4. Create Function:
    a. Click the "Create function" button.

How to setup a VPC

1.

How to add a bastion host (Linux) to the public subnet of a VPC to connect to instances in the private subnet

1.

How to setup IAM so a user can assume an IAM role to access a resource

1.

How to setup AWS Config to monitor resources

1.

How to add inbound rules to both security groups and network ACLs

1.

How to encrypt the root volume of an existing EC2 instance

1.

How to create a SNS topic

1.

How to subscribe to a SNS topic

1.

How to create a CloudWatch alarm using a metrics-based filter

1.