Set up two different groups in IAM

To set up two different groups in AWS with specific access permissions (e.g., backend developers having access to backend servers, and database administrators managing only RDS), you can use AWS Identity and Access Management (IAM). Here's a step-by-step guide:

Step 1: Create IAM Groups

1. **Sign in to the AWS Management Console** and go to **IAM (Identity and Access Management)**.
2. **Create a Group for Backend Developers**:
    - Go to **Groups** on the left panel and select **Create group**.
    - Name the group (e.g., BackendDevelopers).
    - Click **Next** without adding any policies at this point (you'll do this in the next step).
    - Select **Create group**.
3. **Create a Group for Database Administrators**:
    - Repeat the process to create a group named DatabaseAdministrators.

Step 2: Create IAM Policies

Now, create custom policies to restrict resource access for each group.

1. **Create a Policy for Backend Developers**:
    - Go to **Policies** in IAM and select **Create policy**.
    - Choose **JSON** to define permissions, and enter a policy like the following, which grants access to specific EC2 instances:

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeInstances",
    "ec2:StartInstances",
    "ec2:StopInstances",
    "ec2:RebootInstances",
```

```
      "ec2:TerminateInstances"
    ],
    "Resource": [
      "arn:aws:ec2:<region>:<account-id>:instance/<instance-id>"
    ]
   }
 ]
}
```

- Replace <region>, <account-id>, and <instance-id> with the specific details of the instances you want the backend developers to manage.
- Click **Next** and give the policy a name (e.g., BackendServerAccessPolicy).
- Save the policy.

2. **Create a Policy for Database Administrators**:
   - Create another policy with the following permissions to manage RDS instances only:

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "rds:DescribeDBInstances",
    "rds:ModifyDBInstance",
    "rds:DeleteDBInstance",
    "rds:StartDBInstance",
    "rds:StopDBInstance"
   ],
   "Resource": [
    "arn:aws:rds:<region>:<account-id>:db:<db-instance-id>"
   ]
  }
 ]
```

}

- Replace <region>, <account-id>, and <db-instance-id> with specific RDS details.
- Name it (e.g., DatabaseAccessPolicy) and save the policy.

Step 3: Attach Policies to IAM Groups

1. **Attach the Backend Developers Policy**:
   - Go back to **Groups** and select BackendDevelopers.
   - Go to the **Permissions** tab and select **Add permissions** > **Attach policies**.
   - Search for and select BackendServerAccessPolicy.
   - Click **Add permissions**.
2. **Attach the Database Administrators Policy**:
   - Repeat the process for DatabaseAdministrators and attach DatabaseAccessPolicy.

Step 4: Add Users to Groups

1. Go to **Users** and select each user you want to add to a group.
2. In each user's settings, select **Add user to groups**.
3. Choose the appropriate group (e.g., add backend developers to BackendDevelopers, and database administrators to DatabaseAdministrators).

Step 5: Test Access

1. **Verify Access for Backend Developers**:
   - Log in as a backend developer user and check that they can access only the permitted EC2 instances.
2. **Verify Access for Database Administrators**:
   - Log in as a database administrator and check that they can access only the specified RDS instances.

Summary

This setup creates two distinct groups with limited, role-based access to AWS resources, ensuring backend developers only manage backend servers, and database administrators exclusively handle database resources.