Cloud Guru
Hand-On Lab

Introduction to AWS Identity and Access Management (IAM)

Description

AWS Identity and Access Management (IAM) is a service that allows AWS customers to manage users' access and permissions to the AWS accounts and available APIs/services within AWS. IAM can manage users, security credentials (such as API access keys), and allow users to access AWS resources. In this lab, we will walk through the foundations of IAM. We'll focus on user and group management, as well as how to assign access to specific resources using IAM managed policies. We'll learn how to find the login URL where AWS users can log in to their account and explore this from a real-world use case perspective

Objectives

Successfully complete this lab by achieving the following learning objectives:

Add the Users to the Proper Groups

Add the following users to their proper groups:

- user-1 should be in the S3-Support group.
- user-2 should be in the EC2-Support group.
- user-3 should be in the EC2-Admin group.

Use the IAM Sign-in Link to Sign In As a User

Copy the IAM users sign-in link in the AWS console, open an incognito window, and sign in as either user-1, user-2, or user-3 with the password 123456.

Solution

Log in to the live AWS environment using the credentials provided. Make sure you're in the N. Virginia (us-east-1) region throughout the lab.

Lab-Link

Environment Walkthrough

Explore the Users

1. Navigate to IAM > Users.
2. Click user-1.
3. At the top, under Summary, observe the user’s ARN (Amazon Resource Name), path, and creation time.
4. Select the Permissions and Groups tabs, where we'll see user-1 does not have any permissions assigned to it and does not belong to any groups.
5. Select the Security credentials tab to see its access keys, SSH public keys, and HTTPS Git credentials for AWS CodeCommit.
6. Select the Access Advisor tab to see which services the user has accessed and when.
7. Click Users in the left-hand menu, and select user-2 and user-3 to check out their permissions, groups, security credentials, and services.

Explore the Groups

1. Click **Groups** in the left-hand menu.
2. There are three groups we're going to focus on:
   - EC2-Admin: Provides permissions to view, start, and stop EC2 instances
   - EC2-Support: Provides read-only access to EC2
   - S3-Support: Provides read-only access to S3
3. Click any of the groups to see which policy is attached to it.

Note: There are two different kinds of policies for these groups:

- **Managed policies**: Policies shared among users and/or groups that are prebuilt either by AWS or an administrator within the AWS account. When it's updated, the changes to this policy are immediately applied for all users and groups to which it's attached.
- **Inline policies**: Policies assigned to just one user or group that are typically used in one-off situations.

4. Click the **EC2-Admin** group.
5. Click the **Permissions** tab, where we'll see it has a set of permissions associated with it: an inline policy.

6. Click **Show Policy** to see the actions the group is allowed to take (and which resources the action can be taken on) or if it has read-only access. This policy displays JSON access control policy language and provides access on a granular level to AWS resources.
    - From this, we can see we have permission to view, start, and stop EC2 instances on all resources; view all of our elastic load balancers; list metrics; get metric statistics; and describe metrics (which our CloudWatch metrics automatically configured with our EC2 instance). The same permissions apply to our Auto Scaling service. All of these allow us to do it on any resource.
7. Click **Cancel**.
8. Click **Groups** in the left-hand menu.
9. Click the **EC2-Support** group.
10. Click the **Permissions** tab, where we'll see it has a managed policy.
11. Click **Show Policy**.
    - We'll see this group can describe EC2 instances, describe elastic load balancers, describe and list CloudWatch metrics, and describe our autoscaling configurations. What it doesn't allow us to do is stop, start, create, or pretty much anything else. It's essentially read-only, which means we can view what's happening inside EC2, but we can't do anything inside it.
12. Click **Cancel**.
13. Click **Groups** in the left-hand menu.
14. Click the **S3-Support** group.
15. Click the **Permissions** tab. In this scenario, our S3-Support group is only allowed read-only access.
16. Click **Show Policy**, where we'll see the Get and List actions, which allow us to view the objects in an S3 bucket as well as view the S3 bucket itself.
17. Click **Cancel**.

Add the Users to the Proper Groups

Add Users to Groups

1. Still on the S3-Support group page, click the **Users** tab.
2. Click Add **Users to Group**.
3. Select user-1, and click **Add Users**.
4. Click **Groups** in the left-hand menu.
5. Click **EC2-Support**.

6. In the **Users** tab, click **Add Users to Group**.
7. Select user-2, and click **Add Users**.
8. Click **Groups** in the left-hand menu.
9. Click **EC2-Admin**.
10. In the **Users** tab, click **Add Users to Group**.
11. Select user-3, and click **Add Users**.

Use the IAM Sign-in Link to Sign In As a User

Log in as user-1

1. Click **Dashboard** in the left-hand menu.
2. Click the double-papers icon next to the *IAM users sign-in* link at the top.
3. Open an incognito-mode browser window, and paste the link into the browser.
4. Log in as user-1 using the password 123456.
5. Navigate to **S3**.
6. Click **Create bucket**.
7. Enter a random bucket name.
8. Click **Next** > **Next** > **Create bucket**. We'll get an "access denied" error.
9. Close out of the bucket creation box.
10. Navigate to **EC2** > **Instances**. We'll find we don't have the authorized permissions to view any information here because this user doesn't need these permissions for their job role.
11. Log out by clicking **user-1** in the top right corner, and then click **Sign Out**.

Log in as user-2

1. Click the **Sign In to Console** button in the top right corner.
2. Log in as user-2 using the password 123456.
3. Navigate to **EC2** > **Instances**. We should immediately notice user-2 has more permissions than user-1, as we'll be able to view the running instance.
4. With the running instance selected, click **Actions** > **Instance State** > **Stop**.
5. Click **Yes, Stop** in the dialog. We'll then receive a message saying "Error stopping instances," as this user does not have permission to stop instances.
6. Click **Cancel**.
7. Navigate to S3. We'll see an "access denied" error message, as this user doesn't have the right permissions to view anything here.
8. Log out by clicking **user-2** in the top right corner, and then click **Sign Out**.

Log in as user-3

1. Click the **Sign In to Console** button in the top right corner.
2. Log in as user-3 using the password 123456.
3. Navigate to **EC2** > **Instances**.
4. With the running instance selected, click **Actions** > **Instance State** > **Stop**.
5. Click **Yes, Stop** in the dialog. This time, it will let us stop it, and we'll see the instance enter a stopping state. (It will take a few minutes for the instance to finally stop.)
6. Once it’s stopped, click **Actions** > **Instance State** > **Start**.
7. Click **Yes, Start** in the dialog. It will then show a pending status as it moves back into a running status.

Conclusion

Congratulations on successfully completing this hands-on lab!