Lab 1: Working with Amazon Identity and Access Management

Overview	In this lab, you will use the Amazon Identity and Access Management service to create users and roles within an AWS environment. You will then test the permissions of these users and roles to verify that they can only perform the specified actions within the AWS environment.

Objectives	After completing this lab, you will be able to: • Familiarize yourself with the Identity and Access Management (IAM) Console. • Grant permissions to users to use a specific AWS service. • Grant limited permissions to users in a group. • Locate and use the IAM sign-in URL.

Pre-requisites	This lab requires: • Access to a notebook computer with Wi-Fi running Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat). ○ **Note** The qwikLABS lab environment is not accessible using an iPad or tablet device, but you can use these devices to access the student guide. • For Microsoft Windows users: Administrator access to the computer. • An Internet browser such as Chrome, Firefox, or Internet Explorer 9 (previous versions of Internet Explorer are not supported). • An SSH client, such as PuTTY.

Duration	This lab will require around **35 minutes** to complete.

Task 1: Creating IAM Users

Overview	In this part of the lab, you will create two users in IAM, create and manage their passwords and then assign a specific security policy to one of them.

Task 1-1: Create users in IAM	
Overview	In this section of the lab, you will create two users in IAM.

Step	Instruction
1.1.1	In the **AWS Management Console**, click **Identity & Access Management**.
1.1.2	In the **Dashboard** pane, click **Users**.
1.1.3	Click **Create New Users**.
1.1.4	In the first **Enter User Names** box, type **S3TestUser**
1.1.5	In the second **Enter User Names** box, type **EC2TestUser**
1.1.6	If it is not already selected, select the **Generate an access key for each user** option.
1.1.7	Click **Create**.
1.1.8	Click **Download Credentials** to download the users' credentials.
1.1.9	Return to your browser and click **Close**.

Task 1-2: Create a password	
Overview	By default, users that you create do not have access to the AWS Management Console. To grant this access, you need to create a password for the user.

Step	Instruction
1.2.1	Return to the **Users** menu in the IAM Management Console.
1.2.2	Select the **S3TestUser** check box.
1.2.3	In the **User Actions** drop-down list, click **Manage Password**.
1.2.4	If it is not already selected, select the **Assign an auto-generated password** option. Leave the rest as the default. Click **Apply**. A new page confirms that IAM has generated the user's password.
1.2.5	Click **Show User Security Credentials**.
1.2.6	Open the **credentials** file you saved earlier and add a **Password** column to the right of the other three columns.
1.2.7	Paste the password generated by IAM for **S3TestUser** into the **Password** column of the **S3TestUser** entry.
1.2.8	In the IAM Management Console, click **Close**.
1.2.9	IAM asks you to confirm the closing of the window, because you haven't downloaded the user's password. Because you copied the password to an existing file, click **Close** again to close the window. **Note** You could also download this user's credentials at this point, however this would result in having one separate file for each user. For the sake of simplicity, in this lab you will just use one file to store both sets of credentials.
1.2.10	Create a password for **EC2TestUser** by repeating steps **1.2.1** through **1.2.9** (skipping step **1.2.6**) for that user.

Task 1-3: Set permissions	
Overview	In this section of the lab, you will grant the appropriate permissions to your S3TestUser IAM user.

Step	Instruction
1.3.1	Return to the **Users** menu in the IAM Management Console if you are not already there.
1.3.2	From the list of users, click the name of **S3TestUser** to open the **Summary** page.
1.3.3	Click the **Permissions** tab.
1.3.4	Click **Attach Policy**.
1.3.5	Scroll through the list of policies until you locate the entry marked **AmazonS3FullAccess** under the **Policy Name** column. Select the check box for this entry. This policy grants the selected user full access to all Amazon S3 functions. **Note** You can also use the **Search** box at the top of the list to locate this or any policy more easily.
1.3.6	Click **Attach Policy**. Verify that **AmazonS3FullAccess** is listed under **Policy Name** within the **Permissions** group.

Task 2: Creating an IAM Group

Overview	In this part of the lab, you are going to create a group who has full permissions (Start, Stop, Terminate, and so on) with Amazon EC2 instances. Now, instead of attaching a policy directly to each user, you will create a group that has these permissions, and then add a user to that group.

Task 2-1: Create a user group	
Overview	In this section of the lab you will create an IAM group, apply a policy to it, and add the EC2TestUser to it.

Step	Instruction
2.1.1	In the **Dashboard** pane on the left side of the page, click **Groups**.
2.1.2	Click **Create New Group**.
2.1.3	In the **Group Name** box, type **EC2TestGroup**
2.1.4	Click **Next Step**.
2.1.5	Select the **AmazonEC2FullAccess** policy check box from the list. This policy grants any members of the group full access to all Amazon EC2 functions.
2.1.6	Click **Next Step**.
2.1.7	Click **Create Group**.
2.1.8	Select the **EC2TestGroup** check box.
2.1.9	In the **Group Actions** drop-down list, click **Add Users to Group**.
2.1.10	Select **EC2TestUser**, and then click **Add Users**.
2.1.11	In the **Groups** home page, click the name of the **EC2TestGroup** group to display the details of that group.
2.1.12	To verify that **EC2TestUser** has been added to the group, confirm that **EC2TestUser** is listed under **User**.

Task 3: Creating an IAM Role

Overview	In this part of the lab, you will create a role within IAM. A role is an IAM entity that defines a set of permissions for making AWS service requests. IAM roles are not associated with a specific user or group. Instead, roles are assumed by trusted entities such as IAM users, applications, or AWS services such as Amazon EC2. You are going to create an IAM role that allows anyone using that resource to have "describe" permissions to Amazon EC2 instances when it is assigned to a resource. This means that the user can list the Amazon EC2 instances that are running, but cannot start, stop, or otherwise change them.

Task 3-1: Create a role	
Overview	In this section of the lab, you will create an IAM role and add a policy to it.

Step	Instruction
3.1.1	In the **Dashboard** pane, click **Roles**.
3.1.2	Click **Create New Role**.
3.1.3	In the **Role Name** box, type **EC2Describe**
3.1.4	Click **Next Step**.
3.1.5	Under **Select Role Type**, verify that the **AWS Service Roles** option is selected.
3.1.6	Locate **Amazon EC2** at or near the top of the list, and click **Select**.
3.1.7	Select the **AmazonEC2ReadOnlyAccess** policy from the list. This policy grants all entities that assume this role read-only access to Amazon EC2 instances associated with this account. In other words, entities that assume this role will be able to describe and list all existing Amazon EC2 instances, but will not be able to create new instances, or stop or terminate existing instances.
3.1.8	Click **Next Step**.
3.1.9	Click **Create Role**.

Task 4: Testing IAM Users

Overview	You should now have the following: • A user who has full access to only Amazon S3 resources. • A user who has full access to only Amazon EC2 resources. • A role that has read access to only Amazon EC2 resources. Next, you'll test each of these to see how they function. Before you start, obtain the URL associated with your main AWS account for this lab.

Task 4-1: Test the S3 user	
Overview	In this section of the lab, you will test to make sure S3TestUser has the appropriate permissions.

Step	Instruction
4.1.1	In the navigation pane on the left side, click **Dashboard**.
4.1.2	Copy the entire URL displayed below **IAM users sign-in link:**. Paste the URL into a text file and save the file on your local computer.
4.1.3	Open a different web browser such as Firefox or Chrome. **Note** You can open another tab within the same browser; however, a new session will log off **awsstudent**; therefore, you will need to log in again using the password provided by qwikLABS. If you have been using Firefox, using a Chrome or Safari web browser will maintain the session for **awsstudent** while you test **S3TestUser**.
4.1.4	Navigate to the AWS Account Alias URL that you copied in step **4.1.2**.
4.1.5	In the **User Name** box, type **S3TestUser**
4.1.6	In the **Password** box, paste the password for S3TestUser that you saved in your **credentials** file.
4.1.7	Click **Sign In**. The AWS Management Console opens.
4.1.8	Click **EC2**. Because this user does not have any EC2 permissions, messages in the center pane state that you are not authorized to describe various aspects of an EC2 instance.
4.1.9	In the **EC2 Dashboard** pane, click **Instances**. This message appears: *"An error occurred fetching instance data. You are not authorized to perform this operation."* This is because the user that you have used for login has no permissions to Amazon EC2. Next, you will verify whether the user has permissions for Amazon S3.
4.1.10	In the **Services** drop-down list, click **S3**.

4.1.11	Click **Create Bucket**.
4.1.12	In the **Create a Bucket – Select a Bucket Name and Region** dialog box: • In the **Bucket Name** box, type a unique bucket name (e.g., *awsst-lab01-1126*, with no uppercase letters). • In the **Region** drop-down list, select a region you want to create a bucket in. **Note** The bucket name you choose must be unique across all existing bucket names in Amazon S3. One way to help ensure uniqueness is to prefix your bucket names with the name of your organization. Bucket names must comply with certain rules: • Bucket names must be at least 3 characters and no more than 63 characters long. Bucket names can contain lowercase letters, numbers, and hyphens (Note: not uppercase letters). Each label must start and end with a lowercase letter or a number.
4.1.13	Click **Create**.
4.1.14	Click the bucket you just created. The bucket is currently empty.
4.1.15	Click **Upload**.
4.1.16	Click **Add Files**. A file selection dialog box opens.
4.1.17	Select a file from your computer that you want to upload, and then click **Open** (for Windows) or **Choose** (for MacOS).
4.1.18	Click **Start Upload**. The file should upload successfully. This demonstrates that **S3TestUser** has permission to create buckets and upload files to **Amazon S3**.

Task 4-2: Test the EC2 user	
Overview	In this section of the lab, you will test to make sure EC2TestUser has the appropriate permissions.

Step	Instruction
4.2.1	In your current browser window or tab, click **S3TestUser @ *xxxx-xxxx-xxxx***, and from the drop-down list that appears, click **Sign Out**.
4.2.2	Using the AWS Account Alias URL you copied in **Step 1.6.2**, navigate back to your user sign-in page.
4.2.3	In the **User Name** box, type **EC2TestUser**
4.2.4	In the **Password** box, type the password that you saved in the **credentials** file.
4.2.5	Click **Sign In**. The AWS Management Console opens.
4.2.6	Click **EC2**.
4.2.7	In the **EC2 Dashboard** pane, click **Instances**. You should not see any error messages. This demonstrates that **EC2TestUser** has permission to work with Amazon EC2.
4.2.8	On the **Services** menu, click **S3**. Note that you cannot access any S3 resources.

Challenge Task (Optional)

Overview	This part of the lab is a challenge; step-by-step instructions are not provided. Use online documentation to help you if necessary. http://docs.aws.amazon.com/IAM/latest/UserGuide/Using_WorkingWithGroupsAndUsers.html

Challenge	Return to your original browser, where you are logged in as awsstudent. Modify the EC2TestUser's permissions in order to grant them permission to list instance profiles (`iam:ListInstanceProfiles`) as well as pass role (`iam:PassRole`). Then, log back into EC2TestUser and use that account to try to launch an Amazon EC2 instance into the new role that you created. Hint The best way to do this is using Inline Policies rather than Managed Policies. Refer to the online User Guide for AWS Identity and Access Management at http://docs.aws.amazon.com/IAM/latest/UserGuide/PermissionsAndPolicies.html To learn more about granting access to applications on Amazon EC2 instances, see the online documentation at http://docs.aws.amazon.com/IAM/latest/UserGuide/role-usecase-ec2app.html