

## Lesson 07 Demo 02

### Configuring Resource-Based Policy Using Principals

**Objective:** To configure resource-based policies using principals to enable the access to AWS resources

**Tools required:** AWS Management Console

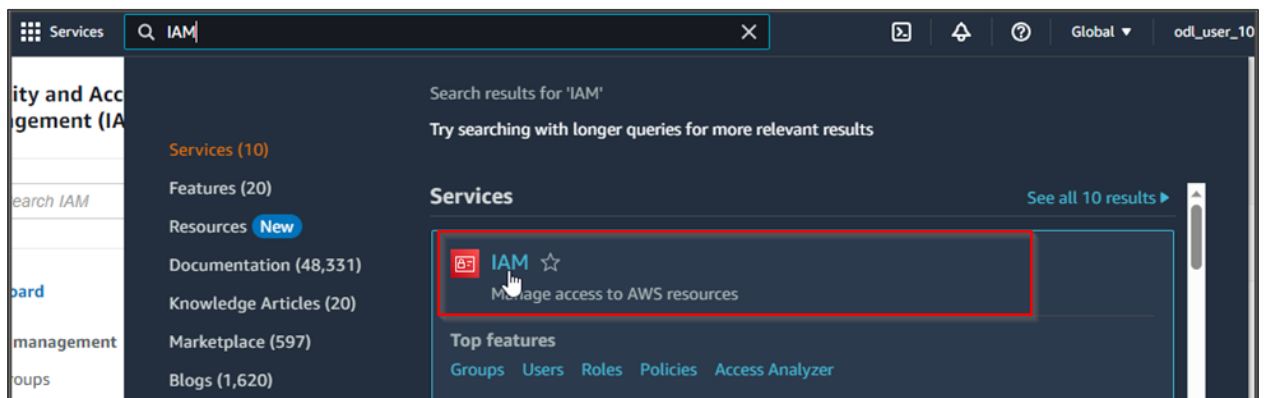
**Prerequisites:** None

Steps to be followed:

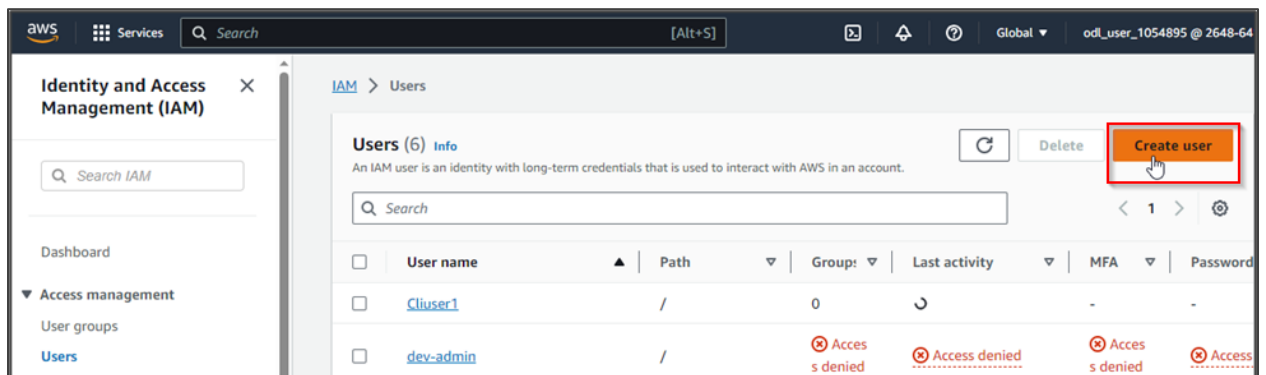
1. Create users and attach policies to them
2. Generate the policy using principals

#### Step 1: Create users and attach policies to them

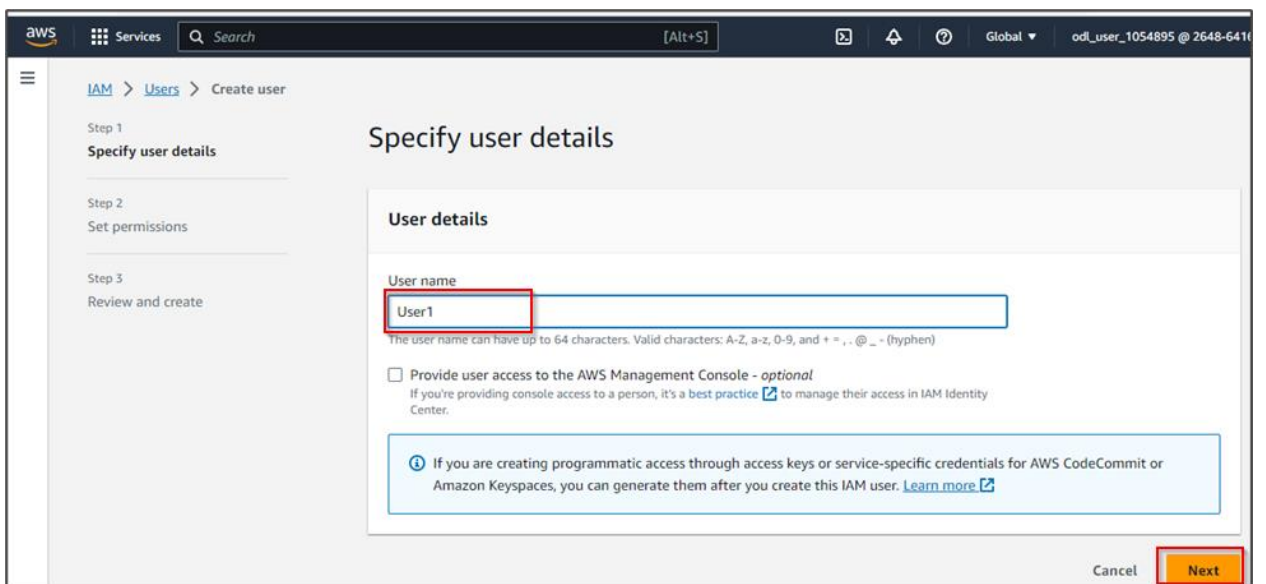
1.1 Navigate to the AWS portal, search for and select **IAM** from the services



1.2 In the IAM dashboard, select **Users** and click on **Create user**



1.3 Provide a name for the user and click on **Next**



1.4 On the **Permissions** page, select **Attach policies directly**. Then, select the **usercreationres** policy and click on **Next**

**Set permissions**

Step 3  
Review and create

**Permissions options**

☐ Add user to group  
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

☐ Copy permissions  
Copy all group memberships, attached managed policies, and inline policies from an existing user.

☒ **Attach policies directly**  
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

**Permissions policies (1/1227)**  
Choose one or more policies to attach to your new user.

Filter by Type  
usercreation X All types 1 match

<input checked="" type="checkbox"/>	Policy name	Type	Attached entities
<input checked="" type="checkbox"/>	<a href="#">usercreationres</a>	Customer managed	1

**Permissions policies (1/1227)**  
Choose one or more policies to attach to your new user.

Filter by Type  
usercreation X All types 1 match

<input checked="" type="checkbox"/>	Policy name	Type	Attached entities
<input checked="" type="checkbox"/>	<a href="#">usercreationres</a>	Customer managed	1

► **Set permissions boundary - optional**

Cancel Previous **Next**

## 1.5 Click on **Create user**

The screenshot shows the 'Review and create' step in the AWS IAM console. The user name is 'User1', the console password type is 'None', and the requirement to reset the password is 'No'. The permissions summary shows a 'UserCreationRestriction' policy. The tags section is empty with an 'Add new tag' button. At the bottom right, the 'Create user' button is highlighted with a red border.

The screenshot shows the AWS IAM console after successful user creation. A green notification banner at the top states 'User created successfully' and provides a 'View user' link. Below the banner, the 'Users (7)' section is visible, showing a list of users. The 'Create user' button is still present.

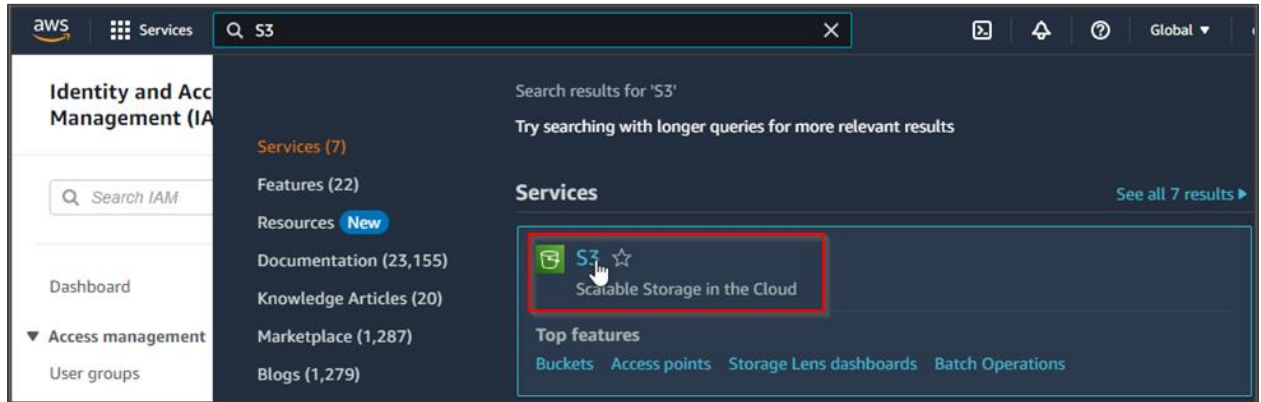
The user is successfully created.

## 1.6 Follow the same steps to create another user

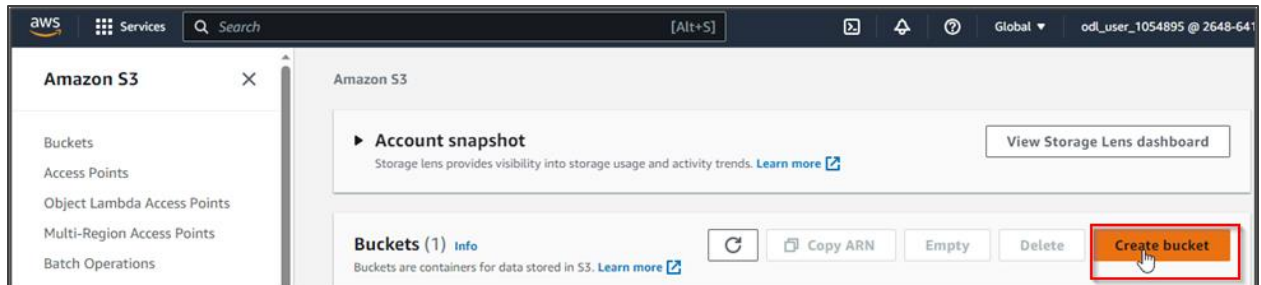
<input type="checkbox"/>	<a href="#">User1</a>	/	0	-	-	-
<input type="checkbox"/>	<a href="#">User2</a>	/	0	-	-	-

## Step 2: Generate the policy using principals

### 2.1 Search for and select S3 from the Services



### 2.2 In the S3 dashboard, click on Create bucket



## 2.3 Provide a name for the bucket and select the AWS Region as given below:

### General configuration

AWS Region  
US East (N. Virginia) us-east-1

Bucket type [Info](#)

☒ **General purpose**  
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory - New**  
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name [Info](#)

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - *optional*  
Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

Services

Search

[Alt+S]

Global

odi\_user\_1054895 @ 2648-64

**Successfully created bucket "user1and2"**  
To upload files and folders, or to configure additional bucket settings choose [View details](#).

View details

Amazon S3 > Buckets

**Account snapshot**  
Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

View Storage Lens dashboard

**Buckets (2)** [Info](#)

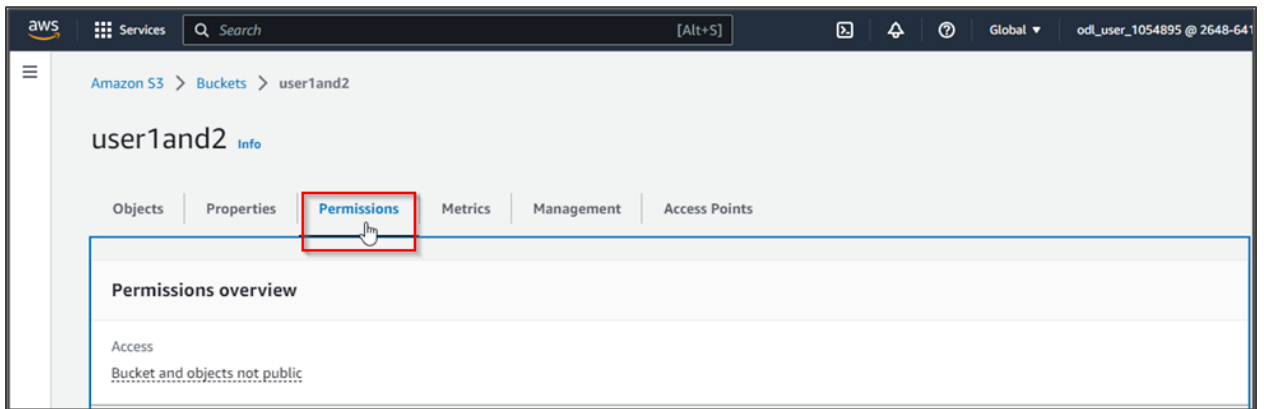
Copy ARN
Empty
Delete
Create bucket

Find buckets by name

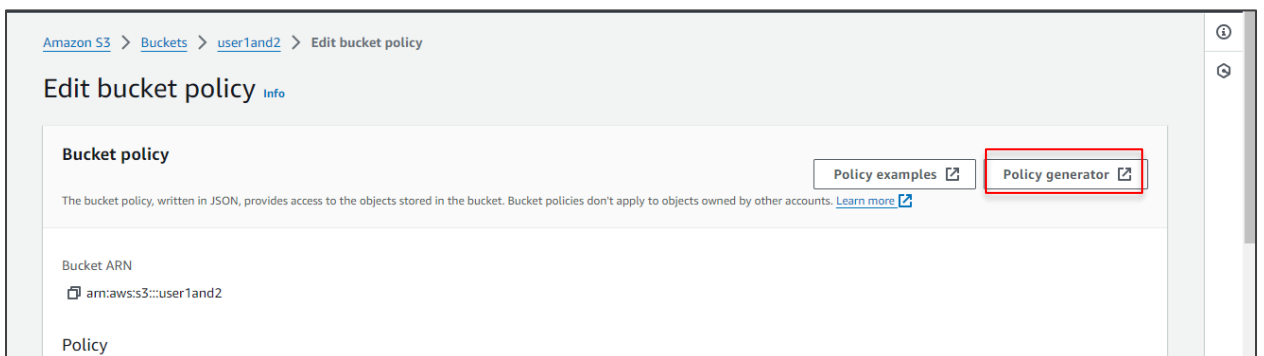
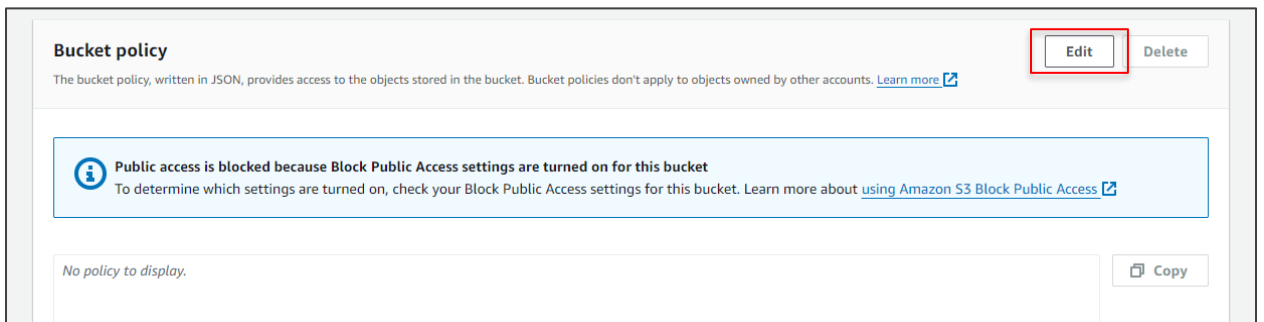
	Name	AWS Region	Access	Creation date
<input type="radio"/>	demoversioningsimplilearn	US East (N. Virginia) us-east-1	Bucket and objects not public	August 31, 2023, 12:08:21 (UTC+05:30)
<input type="radio"/>	user1and2	US East (N. Virginia) us-east-1	Bucket and objects not public	August 31, 2023, 15:05:52 (UTC+05:30)

The bucket is successfully created.

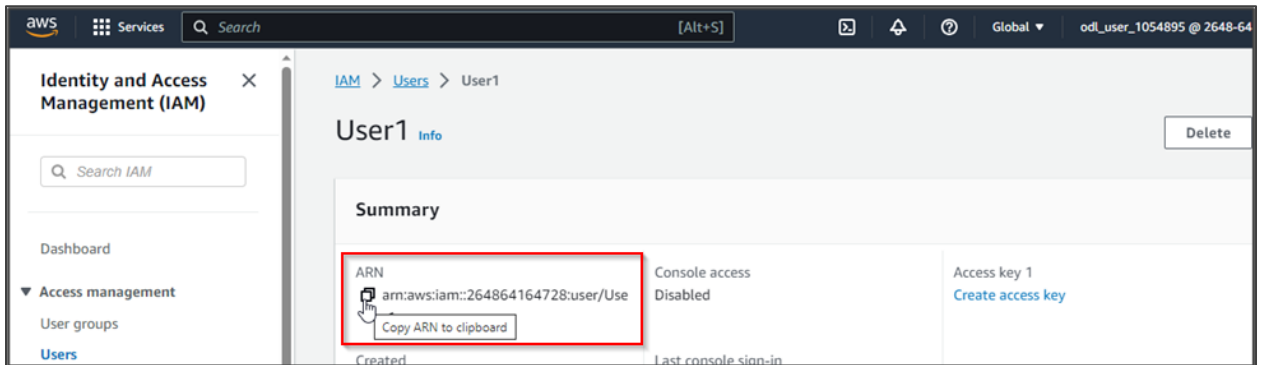
2.4 Select the bucket you created and navigate to the **Permissions** tab as shown:



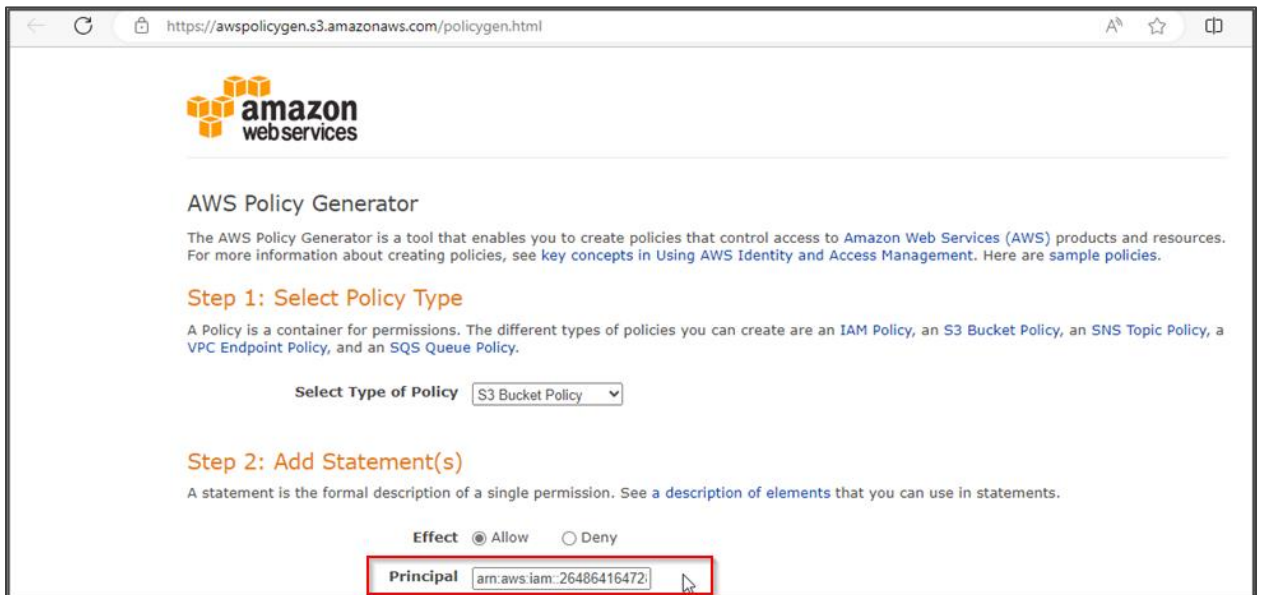
2.5 Click **Edit** under **Bucket policy** section and then click on **Policy generator**



2.6 To enter the value in the principal, go to IAM users, click on **User1**, and then copy the user ARN

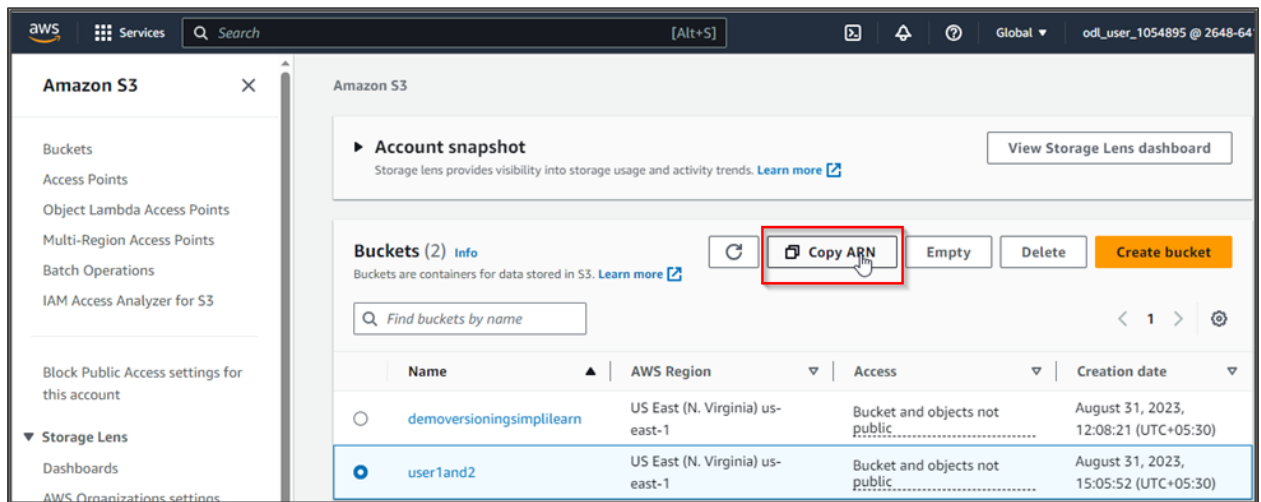


2.7 Paste the copied ARN in the **Principal** tab





2.8 For Amazon Resource Name (ARN), go to the S3 bucket, select the bucket, and click on the **Copy ARN** button



2.9 Paste the copied ARN in the ARN tab and click on **Add Statement**



## 2.10 Now, click on **Generate Policy**

You added the following statements. Click the button below to Generate a policy.

Principal(s)	Effect	Action	Resource	Conditions
• arn:aws:iam::264864164728:user/User1	Allow	s3:*	arn:aws:s3:::user1and2	None

**Step 3: Generate Policy**

A *policy* is a document (written in the [Access Policy Language](#)) that acts as a container for one or more statements.

**Generate Policy** [Start Over](#)

The policy will be generated as shown:

**Policy JSON Document** ✕

Click below to edit. To save the policy, copy the text below to a text editor.  
Changes made below will **not** be reflected in the policy generator tool.

```
{
  "Id": "Policy1693475075264",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1693475060524",
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::user1and2",
      "Principal": {
        "AWS": [
          "arn:aws:iam::264864164728:user/User1"
        ]
      }
    }
  ]
}
```

**Close**

2.11 Copy the JSON and paste it into the bucket policy as shown:

### Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

Edit
Delete

*i*
Public access is blocked because Block Public Access settings are turned on for this bucket

To determine which settings are turned on, check your Block Public Access settings for this bucket. Learn more about using Amazon S3 Block Public Access

```

{
  "Version": "2012-10-17",
  "Id": "Policy1693475075264",
  "Statement": [
    {
      "Sid": "Stmnt1693475060524",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::264864164728:user/User1"
      },
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::user1and2"
    }
  ]
}

```

Copy

The screenshot shows the AWS Management Console interface. At the top, a green notification bar states "Successfully edited bucket policy." The main content area is for the "user1and2" bucket, with the "Permissions" tab selected. The "Permissions overview" section displays "Access" and "Bucket and objects not public". The left sidebar shows the "Amazon S3" navigation menu.

The policy has been successfully updated.

By following these steps, you have successfully configured resource-based policies using principals to enable the access to AWS resources.