

AWS Academy Cloud Security

Foundations

Lab 3.1 Report: Using Resource -Based Policies to Secure an S3 Bucket

TAKI Oussama

3ACI Info

1. Présentation du laboratoire et objectifs

L'objectif principal de ce laboratoire était de comprendre la différence entre les politiques basées sur l'identité (IAM identity-based policies) et les politiques basées sur les ressources, notamment celles appliquées aux buckets S3. Nous avons également examiné la manière dont ces politiques interagissent pour fournir un contrôle d'accès granulaire.

Objectifs clés :

- Comprendre la distinction entre politiques basées sur l'identité et basées sur les ressources.
- Démontrer comment un utilisateur IAM peut assumer un rôle (assume role) afin de modifier ou d'étendre ses permissions.
- Analyser l'impact des différentes politiques sur la visibilité et l'exécution des actions dans la console AWS.

Architecture :

- **Architecture initiale :**

Le laboratoire débute avec un utilisateur IAM nommé *devuser*, appartenant au groupe *DeveloperGroup*.

- **Architecture cible :**

À la fin de l'exercice, nous utilisons des rôles IAM spécifiques (*BucketAccessRole* et *OtherBucketAccessRole*) permettant d'interagir avec des buckets S3 soumis à des restrictions d'accès.

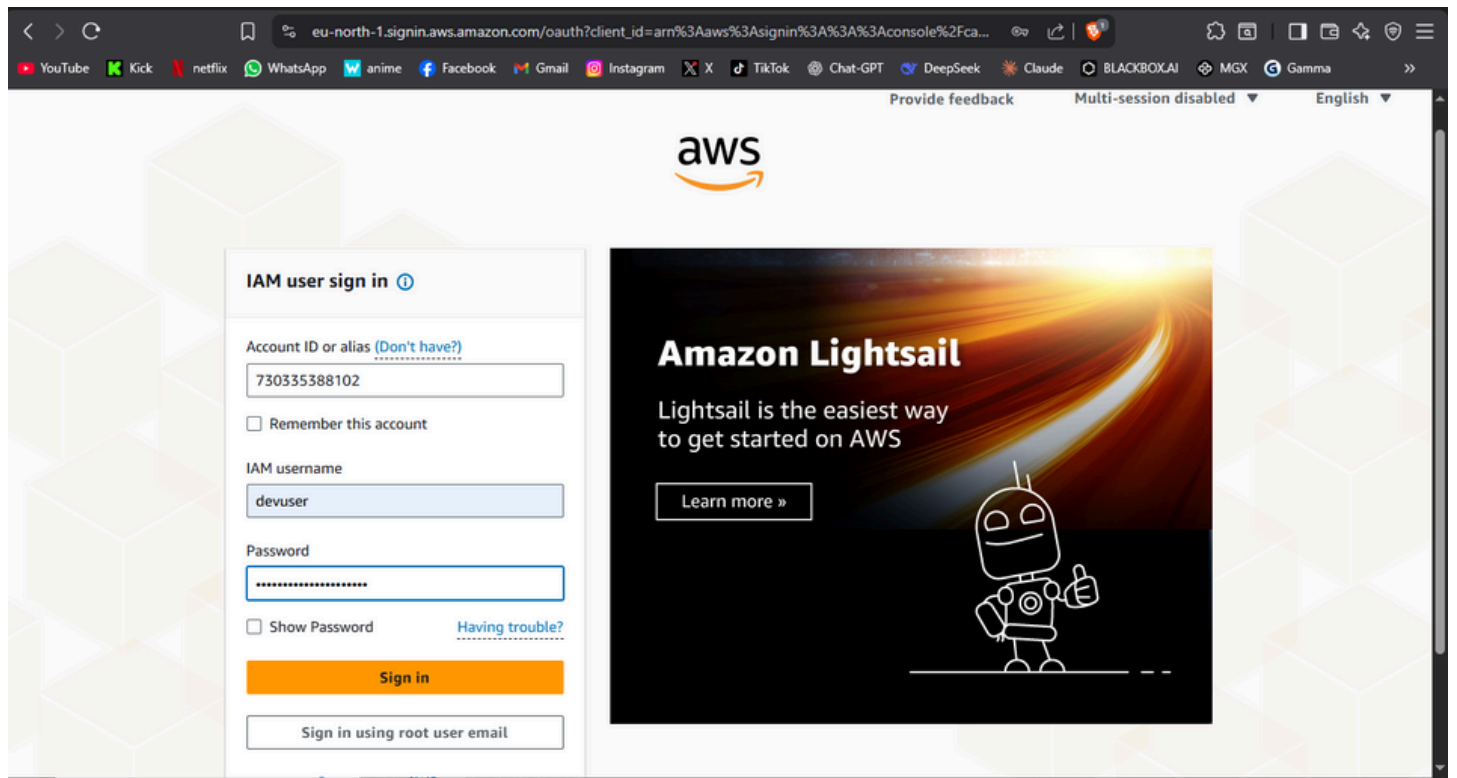
2. Description détaillée des tâches

Tâche 1 : Accès à la console

- **Action :** Connexion à la console de gestion AWS en utilisant les identifiants fournis pour *devuser*.

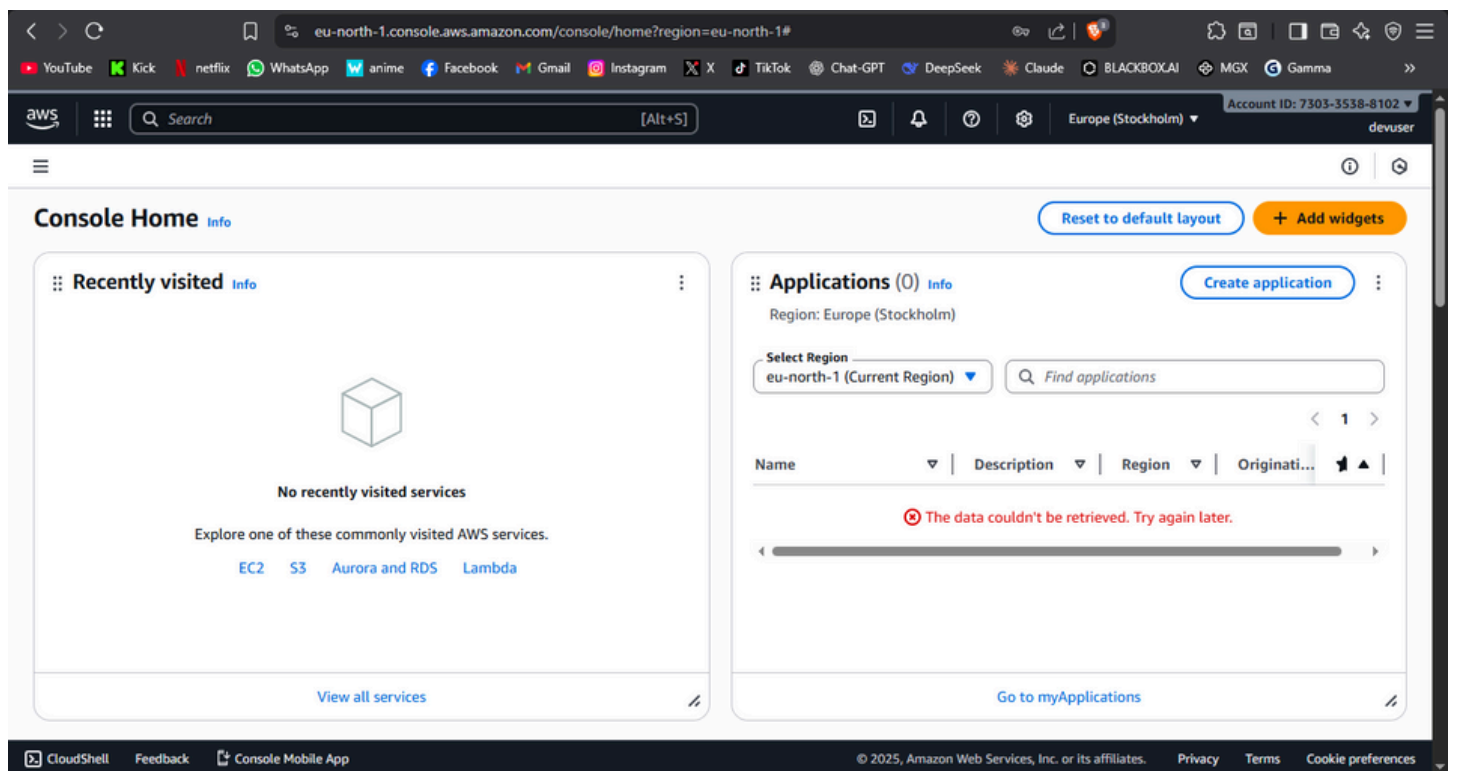
The screenshot displays the AWS Academy interface in a web browser. The address bar shows the URL: `awsacademy.instructure.com/courses/143026/assignments/1668543?module_item_id=13830696`. The page title is "Lab 3.1: Using Resource-Based Policies to Secure an S3 Bucket". The left sidebar contains navigation links: Home, Modules, Discussions, Grades, Lucid (Whiteboard), Account, Dashboard, Courses, Calendar, Inbox, History, and Help. The main content area is titled "Lab 3.1: Using Resource-Based Policies to Secure an S3 Bucket" and includes a "Lab overview and objectives" section. The objectives section states: "In this lab, you will learn how to configure permissions by using AWS Identity and Access Management (IAM) identity-based and resource-based policies, such as Amazon Simple Storage Service (Amazon S3) bucket policies. You will also learn how". On the right side, there is a "Lab Details" panel showing the accumulated lab time as 02:04:00 (124 minutes). It also displays the status "No running instance" and provides links for "SSH key", "Download PEM", "Download PPK", and "AWS SSO Download URL". A table lists the following details:

Field	Value
IAMUserPassword	lgw-09480c139c0ff28e7
AccountID	730335388102
IAMUserLoginURL	https://730335388102.signin.aws.amazon.com/console
Region	us-east-1



Observation :

La connexion s'est effectuée avec succès, nous amenant directement sur la page d'accueil de la console AWS.

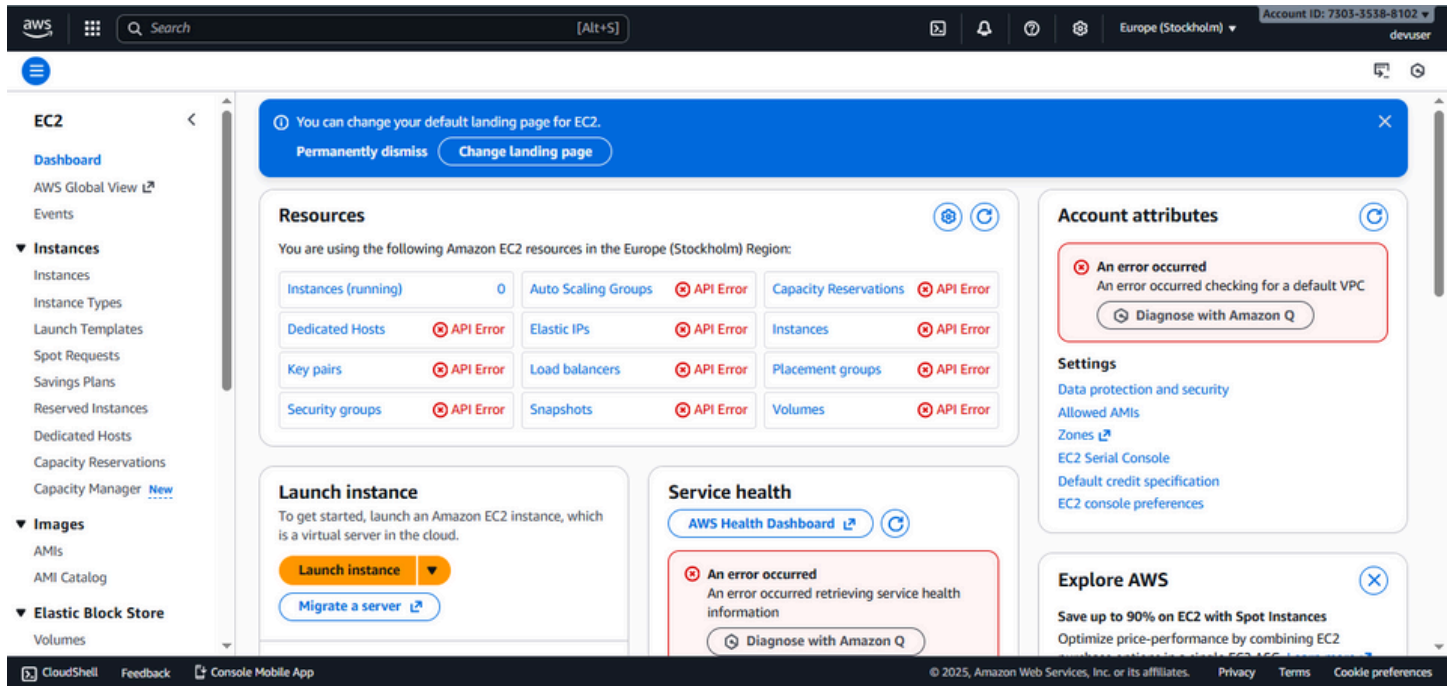


Tâche 2 : Tentative d'accès en lecture aux services AWS

Action 1 : Accéder à Amazon EC2

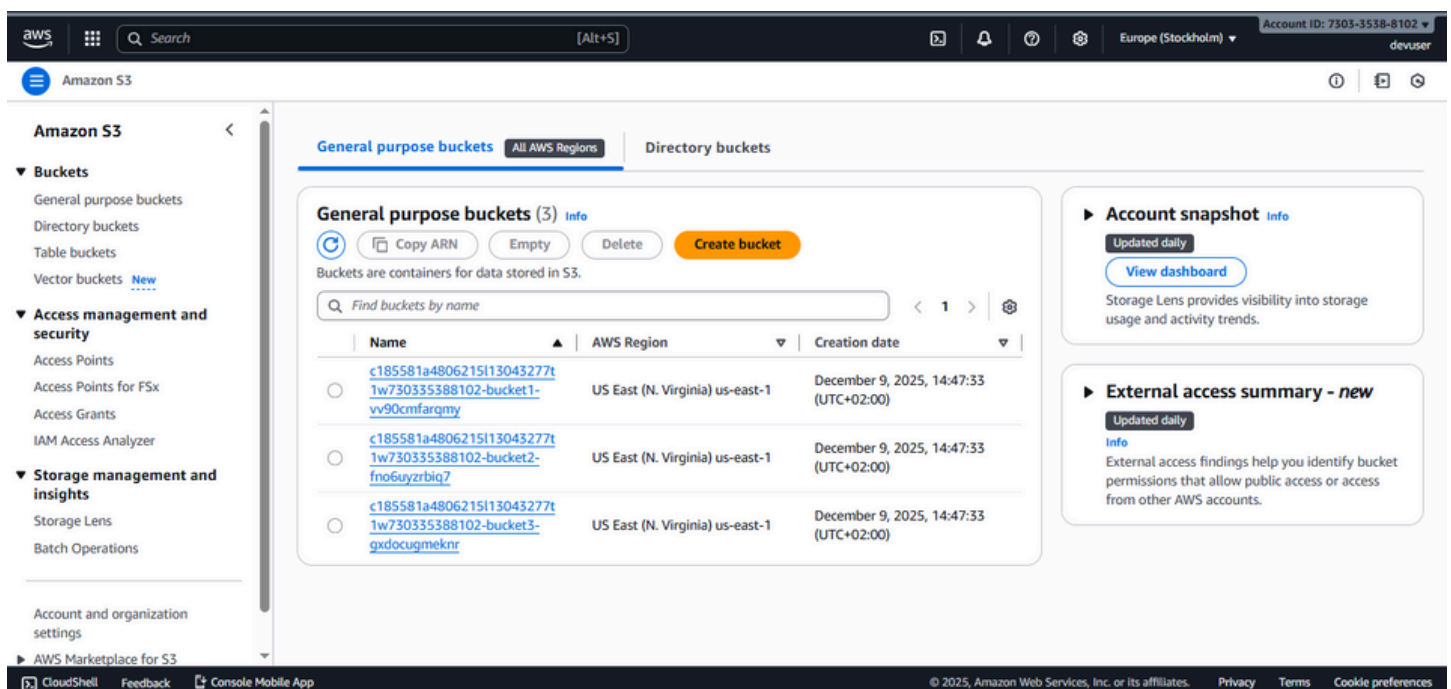
- **Étape** : Navigation vers le tableau de bord EC2 pour consulter les instances.

- **Observation** : Plusieurs messages d'« API Error » sont apparus. Il était impossible d'afficher la liste des instances, de consulter les types d'instances ou d'accéder à toute ressource EC2
- **Raison** :
L'utilisateur *devuser* ne possède pas les permissions nécessaires — comme *ec2:DescribeInstances* — dans sa politique IAM, ce qui empêche l'accès aux ressources EC2.



Action 2 : Accès à Amazon S3

- **Étape** : Navigation vers la console Amazon S3.

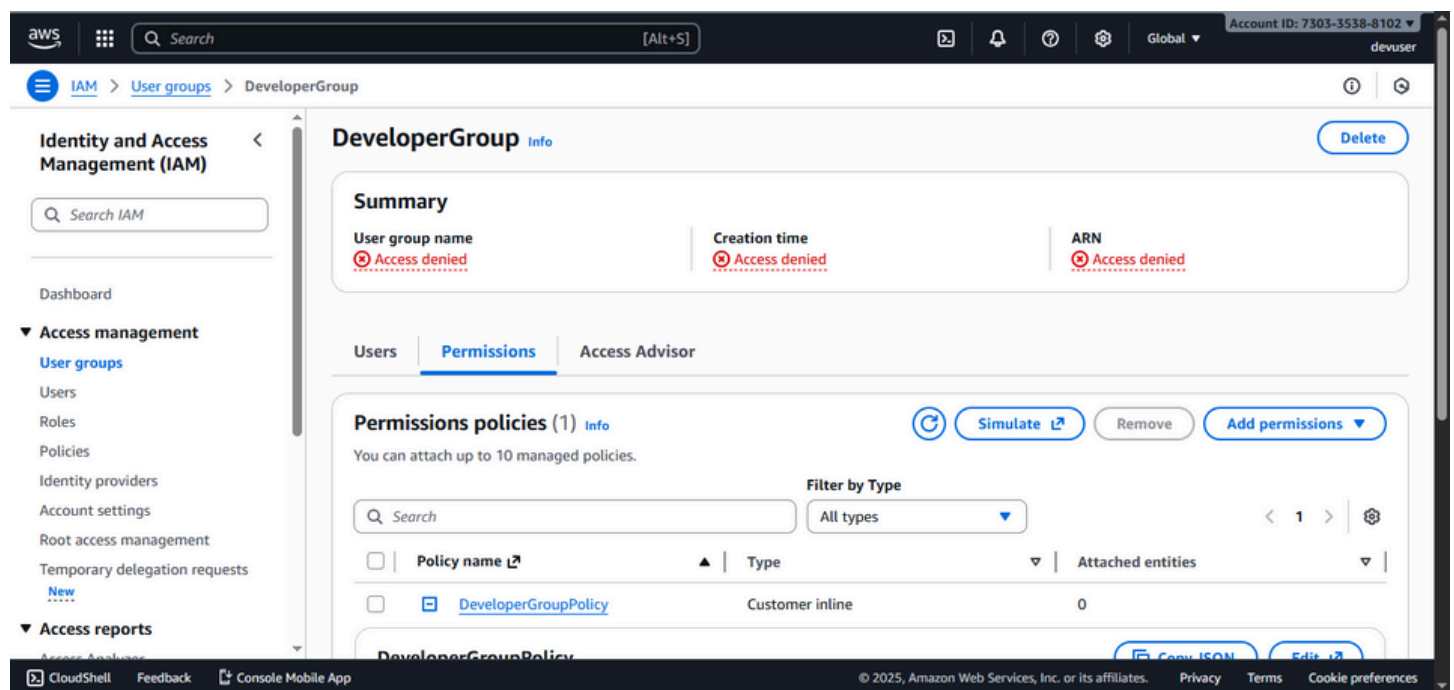


- **Observation** : La liste des buckets (*bucket1*, *bucket2*, *bucket3*) était visible, mais la colonne « Access » affichait « Insufficient permissions » ou « Error ».
- **Raison** : L'utilisateur *devuser* dispose de la permission *ListAllMyBuckets*, ce qui lui permet de voir les noms des buckets. Cependant, il ne possède pas les autorisations nécessaires pour consulter la configuration ou les paramètres d'accès public de chaque bucket.

Tâche 3 : Analyse de la politique basée sur l'identité

Action :

- Accès à la console IAM → *Users* → *devuser* → *Groups* → *DeveloperGroup* → *Permissions*.



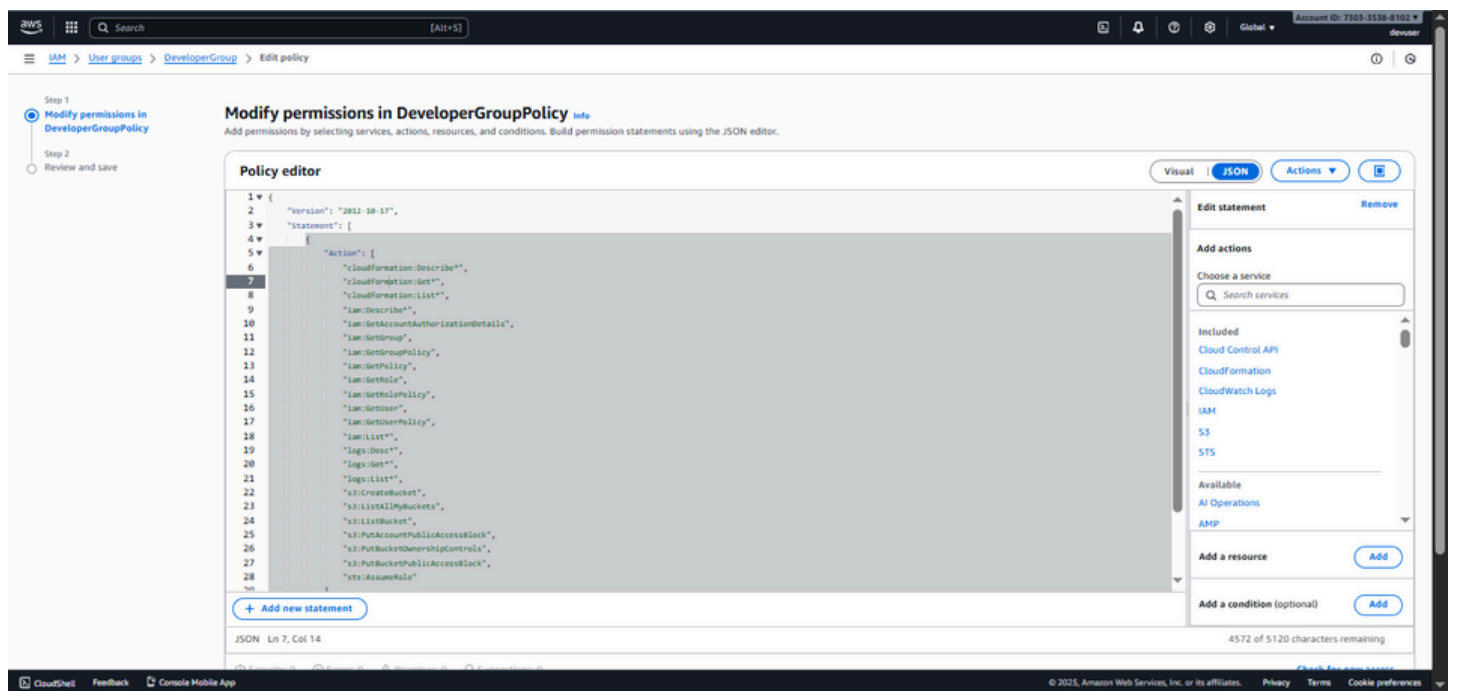
- Ouverture de la politique *DeveloperGroupPolicy* afin d'examiner le fichier JSON.

Analyse de la politique :

La politique associée au groupe est une **Identity-Based Policy** (politique basée sur l'identité).

- **Actions autorisées** : *s3:ListAllMyBuckets*, *s3>CreateBucket*, *s3:ListBucket*.
- **Actions manquantes ou non autorisées** : Absence d'autorisations *ec2:* (ce qui explique les erreurs lors de l'accès à EC2) et absence de permission *s3:PutObject* au niveau global.

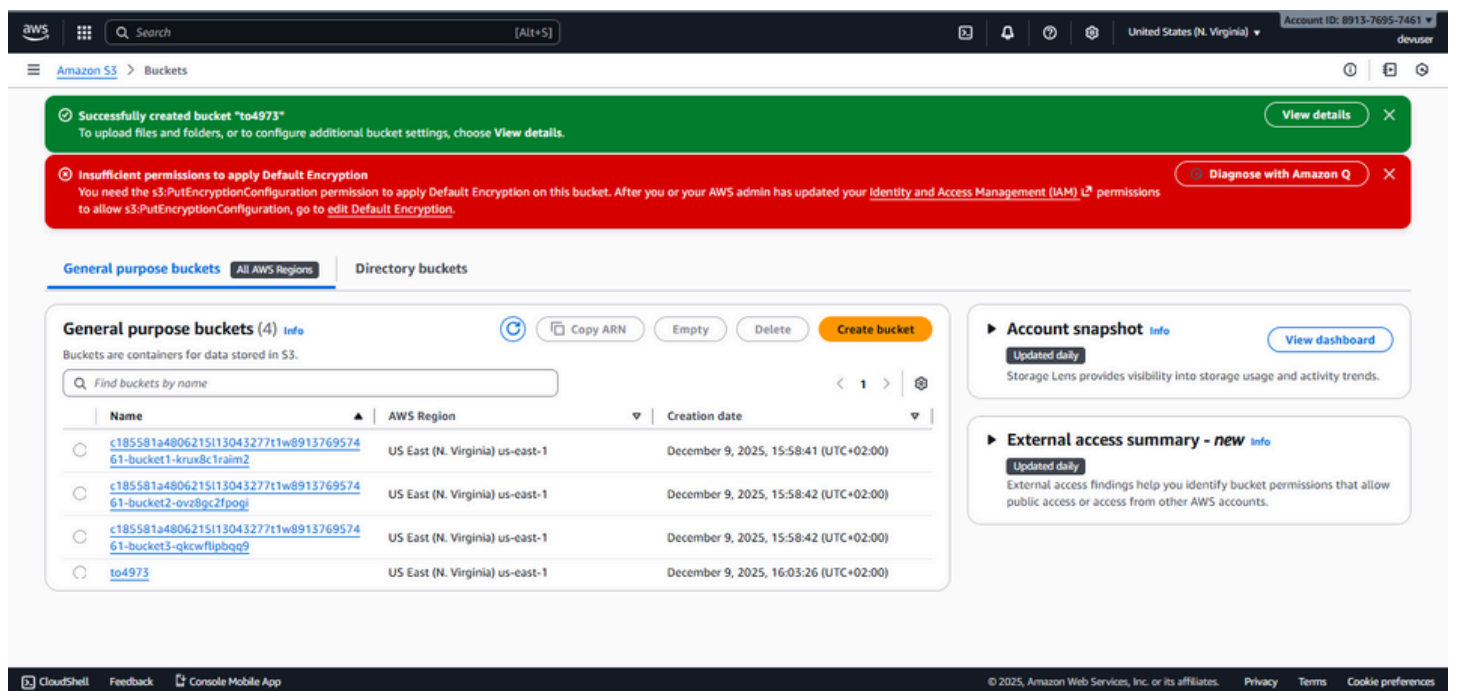
Cela explique pourquoi nous pouvions voir les noms des buckets (permission de liste), mais pas exécuter d'autres actions administratives.



Tâche 4 : Tentative d'accès en écriture aux services AWS

Action 1 : Création d'un bucket

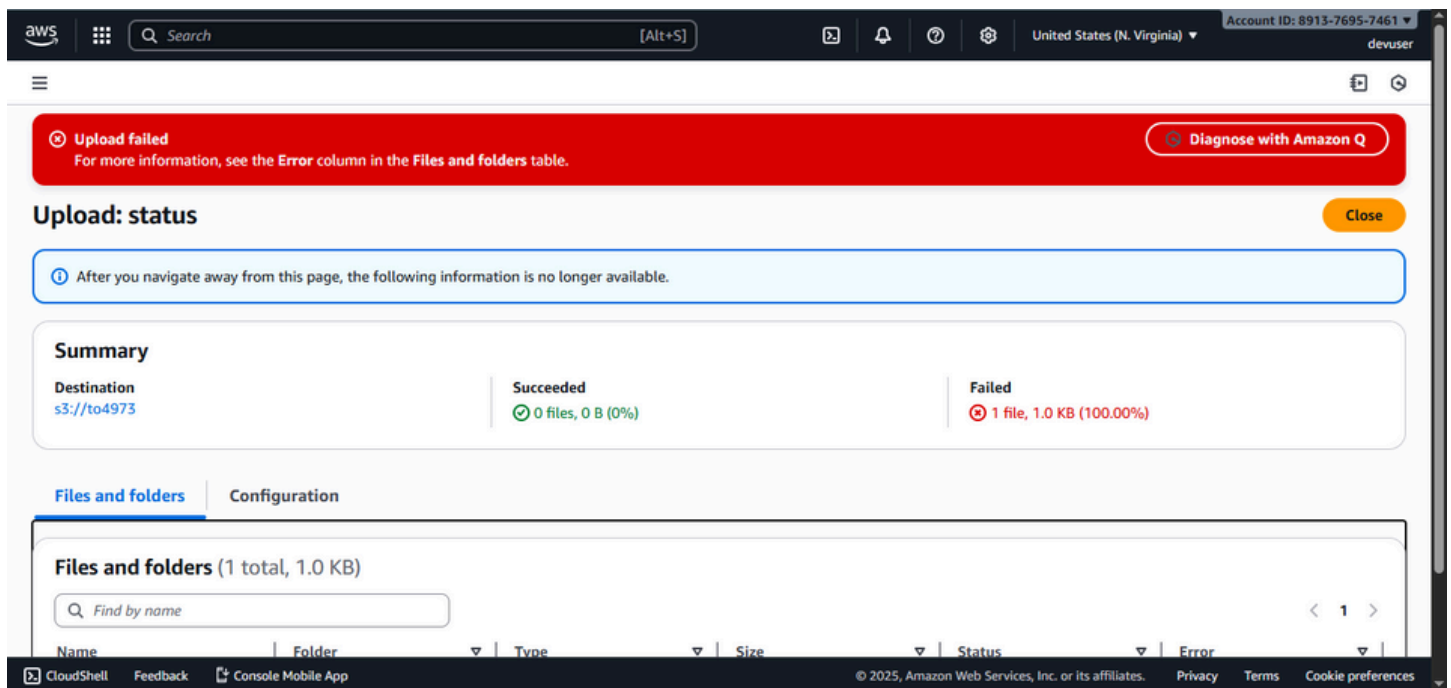
- **Étape** : Clic sur « Create bucket » puis création d'un bucket nommé *to4973*.



- **Résultat** : Succès.
- Raison : La politique *DeveloperGroupPolicy* autorise explicitement l'action *s3:CreateBucket*.

Action 2 : Téléversement d'un objet

- **Étape** : Ouverture du bucket nouvellement créé, puis tentative de téléverser *Image1.jpg*.
- **Résultat** : Échec (message *Upload failed*).
- **Raison** : Bien que nous soyons propriétaires du bucket que nous venons de créer, la politique IAM associée à *devuser* ne contient pas l'autorisation *s3:PutObject*. Dans AWS, la permission de créer une ressource n'implique pas automatiquement la permission de modifier son contenu si la politique reste restrictive.



Tâche 5 : Assumer un rôle IAM

Test préalable : tentative de téléchargement avant d'assumer le rôle

- **Action** : Navigation vers *bucket1* puis tentative de téléchargement d'un fichier existant.
- **Résultat** : Échec (erreur *Access Denied*).

Amazon S3 > Buckets > c185581a4806215113043277t1w891376957461-bucket1-kruX8c1raim2

Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Name	Type	Last modified	Size	Storage class
Image1.jpg	jpg	December 9, 2025, 15:59:13 (UTC+02:00)	1.1 MB	Standard
Image2.jpg	jpg	December 9, 2025, 15:59:15 (UTC+02:00)	375.4 KB	Standard

Image2.jpg

Object overview

Owner
63af3c4c9fca9d8e4dadf100cbc51518983187f0e1e00e808c85d2036831aee7

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

Last modified
December 9, 2025, 15:59:15 (UTC+02:00)

Size
375.4 KB

Type
jpg

S3 URI
<s3://c185581a4806215113043277t1w891376957461-bucket1-kruX8c1raim2/Image2.jpg>

Amazon Resource Name (ARN)
<arn:aws:s3:::c185581a4806215113043277t1w891376957461-bucket1-kruX8c1raim2/Image2.jpg>

Entity tag (Etag)
<a3ae359c8a8fa2fe046e2297c458c8a5>

Object URL
<https://c185581a4806215113043277t1w891376957461-bucket1-kruX8c1raim2.s3.us-east-1.amazonaws.com/Image2.jpg>

This XML file does not appear to have any style information associated with it. The document tree is shown below.

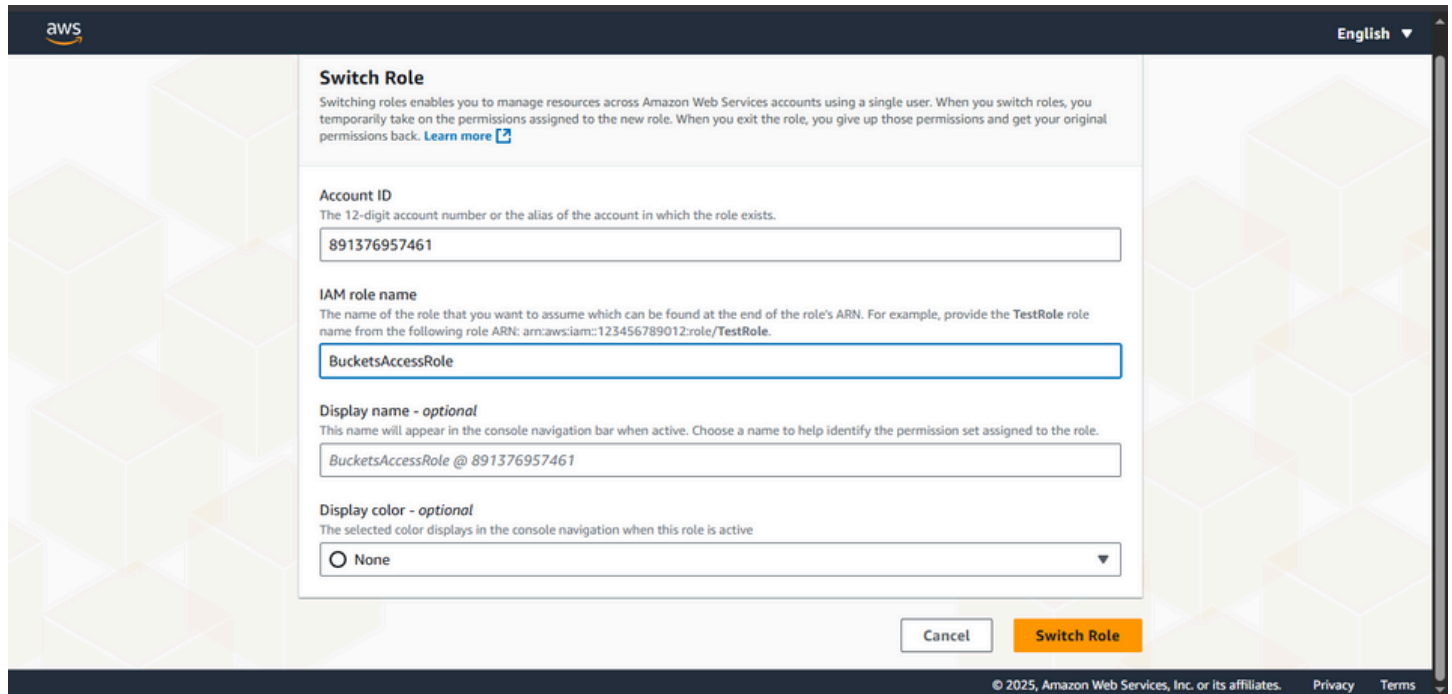
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>Request has expired</Message>
  <X-Amz-Expires>300</X-Amz-Expires>
  <Expires>2025-12-09T13:30:11Z</Expires>
  <ServerTime>2025-12-09T14:24:34Z</ServerTime>
  <RequestId>G9RCX7C44REST0FY</RequestId>
  <HostId>aePwISe4yrqinMcJEAzwhwYMXG6JazAS/vFUM2Vy7BnWv1t0BRbVKnL2165k+4NSX+6Y0uM45nQ=</HostId>
</Error>
```

Raison :

La politique basée sur l'identité de *devuser* (*DeveloperGroupPolicy*) autorise *s3:ListAllMyBuckets* et *s3:ListBucket*, mais ne contient pas la permission *s3:GetObject*, indispensable pour télécharger un objet depuis un bucket. Cela confirme que les

permissions par défaut de l'utilisateur limitent l'accès en lecture au contenu des objets, même s'il peut voir la liste des buckets.

Action : (tu peux m'envoyer la suite pour la reformuler)



The screenshot shows the AWS IAM 'Switch Role' dialog box. At the top, there's a title 'Switch Role' and a brief explanation. Below this, there are four input fields: 'Account ID' (containing '891376957461'), 'IAM role name' (containing 'BucketsAccessRole'), 'Display name - optional' (containing 'BucketsAccessRole @ 891376957461'), and 'Display color - optional' (set to 'None'). At the bottom right, there are 'Cancel' and 'Switch Role' buttons. The background of the console shows a hexagonal pattern and the user's current identity 'BucketsAccessRole'.

Switch Role
Switching roles enables you to manage resources across Amazon Web Services accounts using a single user. When you switch roles, you temporarily take on the permissions assigned to the new role. When you exit the role, you give up those permissions and get your original permissions back. [Learn more](#)

Account ID
The 12-digit account number or the alias of the account in which the role exists.

IAM role name
The name of the role that you want to assume which can be found at the end of the role's ARN. For example, provide the `TestRole` role name from the following role ARN: `arn:aws:iam::123456789012:role/TestRole`.

Display name - optional
This name will appear in the console navigation bar when active. Choose a name to help identify the permission set assigned to the role.

Display color - optional
The selected color displays in the console navigation when this role is active.
☐ None

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#)

Action :

- Ouverture du menu utilisateur en haut à droite puis clic sur Switch Role.
- Saisie de l'ID du compte ainsi que du nom du rôle : `BucketsAccessRole`.
- Changement visible : L'identité affichée en haut à droite a été remplacée par *BucketsAccessRole*, confirmant que le rôle a bien été assumé.

Test d'accès : (envoie-moi la suite et je te la reformule)

Account ID: 8913-7695-7461

BucketsAccessRole @ 891376957461

United States (N. Virginia)

Amazon S3

Buckets

c185581a4806215113043277t1w891376957461-bucket1-krux8c1raim2

Amazon S3

Buckets

General purpose buckets

Directory buckets

Table buckets

Vector buckets

Access management and security

Access Points

Access Points for FSx

Access Grants

IAM Access Analyzer

Storage management and insights

Storage Lens

Batch Operations

c185581a4806215113043277t1w891376957461-bucket1-krux8c1raim2

Info

Objects

Metadata

Properties

Permissions

Metrics

Management

Access Points

Objects (1/2)

Copy S3 URI

Copy URL

Download

Open

Delete

Actions

Create folder

Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

Show versions

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	Image1.jpg	jpg	December 9, 2025, 15:59:13 (UTC+02:00)	1.1 MB	Standard
<input checked="" type="checkbox"/>	Image2.jpg	jpg	December 9, 2025, 15:59:15 (UTC+02:00)	375.4 KB	Standard

You can attach up to 10 managed policies.

Filter by Type: All types

Policy name	Type	Attached entities
GetBucketPolicy	Customer inline	0
GrantBucket1Access	Customer inline	0
ListAllBucketsPolicy	Customer inline	0

ListAllBucketsPolicy [Copy JSON](#) [Edit](#)

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Action": [
6         "s3:ListAllMyBuckets"
7       ],
8       "Resource": "*",
9       "Effect": "Allow"
10    }
11  ]
12 }
```

1. **Bucket 1** : Le téléchargement des objets a fonctionné sans problème.
2. **Bucket 2** : L'upload d'objets était autorisé, car le rôle disposait de la permission `s3:PutObject` spécifiquement sur `bucket2`.

Upload succeeded
For more information, see the Files and folders table.

Upload: status [Close](#)

After you navigate away from this page, the following information is no longer available.

Summary

Destination	Succeeded	Failed
s3://c185581a48062151130432771w449039445877-bucket2-ylsfa5cpevhb	1 file, 375.4 KB (100.00%)	0 files, 0 B (0%)

[Files and folders](#) [Configuration](#)

Files and folders (1 total, 375.4 KB)

Find by name

Name	Folder	Type	Size	Status	Error
image2.jpg	-	image/jpeg	375.4 KB	Succeeded	-

Analyse :

En assumant le rôle, nous avons temporairement abandonné les permissions de `devuser` pour adopter celles de `BucketsAccessRole`. Ce rôle possédait une politique dédiée autorisant la lecture et/ou l'écriture sur certains buckets, des permissions que `devuser` ne détenait pas initialement.

Voici la reformulation :

Tâche 6 : Comprendre les politiques basées sur les ressources

Action :

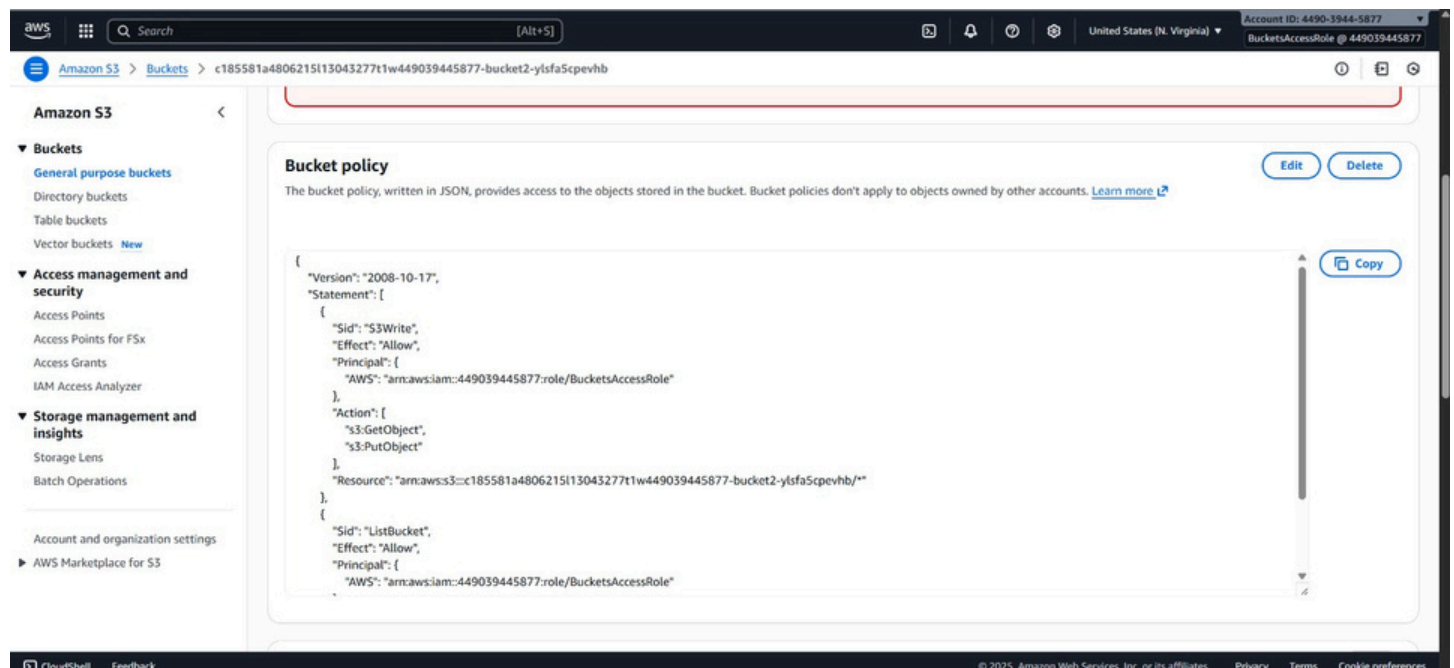
- En étant connecté avec le rôle *bucket-access-role*, nous avons navigué vers *Bucket 2* puis ouvert l'onglet **Permissions**.
- Nous avons consulté la **Bucket Policy**.

Analyse de la politique :

Il s'agit d'une **Resource-Based Policy**, c'est-à-dire une politique attachée directement à la ressource (le bucket S3), contrairement aux politiques basées sur l'identité qui sont liées à un utilisateur ou à un rôle IAM.

- La politique mentionnait explicitement *BucketsAccessRole* comme **Principal** autorisé.
- Cela confirme que l'accès à une ressource peut être accordé soit par la politique de l'utilisateur (Identity-Based Policy), soit par la politique de la ressource (Resource-Based Policy) lorsqu'il s'agit du même compte.
- Dans un contexte inter-comptes (cross-account), les deux types de politiques sont souvent nécessaires, mais dans ce laboratoire l'objectif principal était d'observer comment le rôle obtenait l'autorisation via la bucket policy.

Si tu veux, envoie-moi la dernière partie (conclusion, résumé, etc.) et je la reformule aussi.



3. Challenge Task: : Téléversement dans Bucket 3

Objectif : Envoyer *Image2.jpg* dans *bucket3*.

Tentative 1 : En tant que *devuser*

Upload failed

For more information, see the Error column in the Files and folders table.

Diagnose with Amazon Q

Upload: status

Close

After you navigate away from this page, the following information is no longer available.

Summary

Destination

s3://c185581a4806215113043277t1w449039445877-bucket3-cay0am8i935d

Succeeded

0 files, 0 B (0%)

Failed

1 file, 375.4 KB (100.00%)

Files and folders

Configuration

Files and folders (1 total, 375.4 KB)

Find by name

Name	Folder	Type	Size	Status	Error
Image2 (1).jpg	-	image/jpeg	375.4 KB	Failed	Access denied

Access denied

Diagnose with Amazon Q

API response

User: arn:aws:iam::449039445877:user/devuser is not authorized to perform: s3:GetBucketPublicAccessBlock on resource: "arn:aws:s3::c185581a4806215113043277t1w449039445877-bucket3-cay0am8i935d" because no identity-based policy allows the s3:GetBucketPublicAccessBlock action

Bucket policy

Edit

Delete

You don't have permission to get bucket policy

Diagnose with Amazon Q

You or your AWS administrator must update your IAM permissions to allow s3:GetBucketPolicy. After you obtain the necessary permission, refresh the page. Learn more about [identity and access management in Amazon S3](#)

API response

Object Ownership

Edit

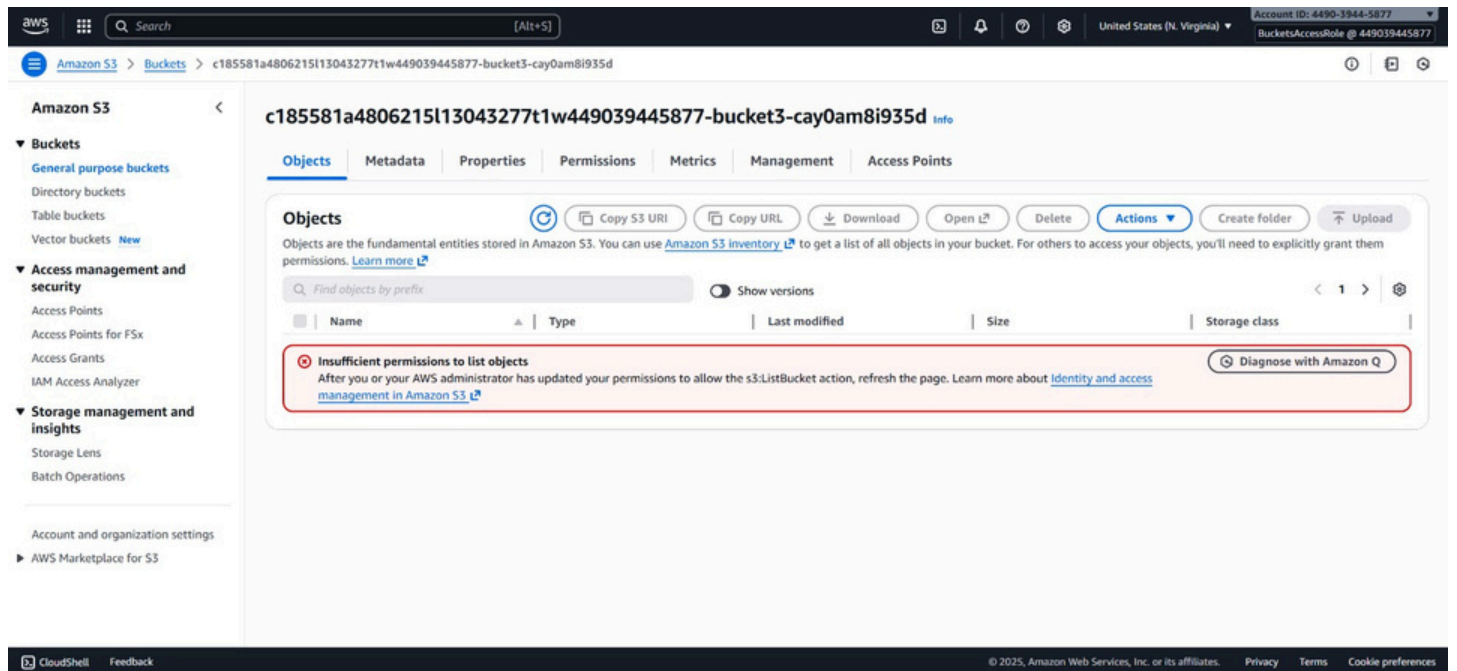
You don't have permission to view Object ownership (bucket settings) configuration

Diagnose with Amazon Q

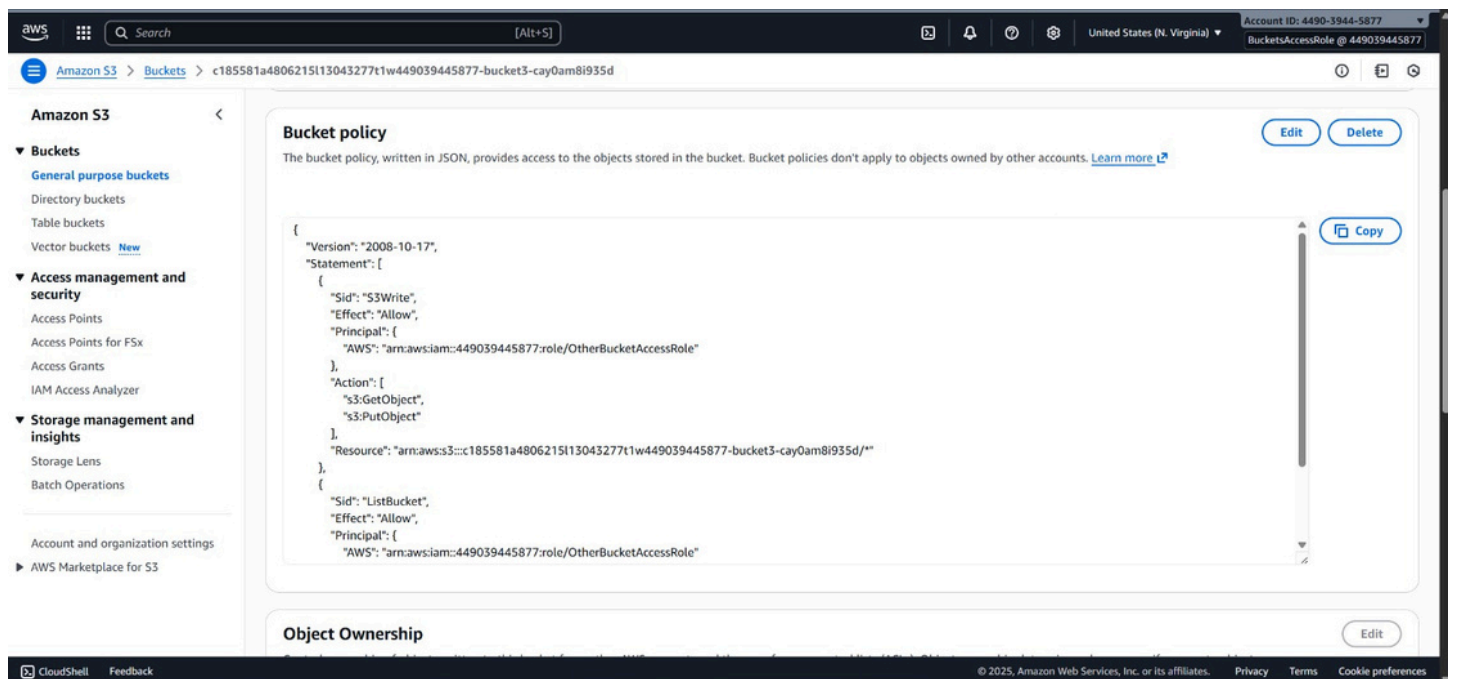
You need s3:GetBucketOwnershipControls to view Object ownership (bucket settings) configuration. Learn more about [Object ownership in Amazon S3](#)

API response

Tentative 2 : En utilisant le rôle *BucketAccessRole*



Résultat : Échec. La politique associée au rôle ne contenait aucune permission permettant d'interagir avec *bucket3*. Toutefois, nous pouvions consulter la bucket policy.



Solution :

1. Nous avons identifié un second rôle disponible dans le compte : **OtherBucketAccessRole**.
2. Nous avons de nouveau utilisé la fonctionnalité **Switch Role** pour passer du rôle *BucketsAccessRole* au rôle *OtherBucketAccessRole*.

Switch Role

Switching roles enables you to manage resources across Amazon Web Services accounts using a single user. When you switch roles, you temporarily take on the permissions assigned to the new role. When you exit the role, you give up those permissions and get your original permissions back. [Learn more](#)

Account ID
The 12-digit account number or the alias of the account in which the role exists.

449039445877

IAM role name
The name of the role that you want to assume which can be found at the end of the role's ARN. For example, provide the `TestRole` role name from the following role ARN: `arn:aws:iam::123456789012:role/TestRole`.

OtherBucketsAccessRole

Display name - optional
This name will appear in the console navigation bar when active. Choose a name to help identify the permission set assigned to the role.

OtherBucketsAccessRole @ 449039445877

Display color - optional
The selected color displays in the console navigation when this role is active

☐ None

Cancel Switch Role

1. Nous avons navigué vers *bucket3*.
2. Nous avons cliqué sur **Upload**, sélectionné *Image2.jpg*, puis validé l'envoi.
3. **Résultat** : Succès.

Explication :

Le rôle **OtherBucketAccessRole** possédait une politique autorisant explicitement l'action *s3:PutObject* sur la ressource *bucket3*. En assumant ce rôle spécifique, nous avons obtenu les permissions nécessaires pour réaliser la tâche.

Upload succeeded
For more information, see the [Files and folders](#) table.

Upload: status Close

After you navigate away from this page, the following information is no longer available.

Summary

Destination
s3://c185581a4806215f13043277e1w449039445877-bucket3-cay0am8i935d

Succeeded
1 file, 375.4 KB (100.00%)

Failed
0 files, 0 B (0%)

Files and folders | Configuration

Files and folders (1 total, 375.4 KB)

Find by name

Name	Folder	Type	Size	Status	Error
image2.jpg	-	image/jpeg	375.4 KB	Succeeded	-

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Total score

15/15

TASK 4 - Create bucket

5/5

TASK 5 - Uploaded object

5/5

CHALLENGE TASK - Uploaded object

5/5

4. Conclusion

Ce laboratoire a clairement démontré la flexibilité et la granularité du contrôle d'accès dans AWS :

1. **Les politiques d'identité** associées à *devuser* offraient un accès de base, mais limitaient les actions sensibles, comme la création d'instances EC2 ou l'écriture dans des buckets non autorisés.
2. **Les rôles IAM** nous ont permis d'élever temporairement nos permissions afin d'accomplir des tâches spécifiques (gestion ciblée de certains buckets) sans avoir à attribuer ces privilèges de manière permanente à l'utilisateur.
3. **Les politiques basées sur les ressources**, appliquées directement aux buckets S3, ont ajouté une couche supplémentaire de contrôle, en définissant précisément quels *principals* (utilisateurs ou rôles) étaient autorisés à interagir avec les données du bucket.