AWS Documentation

# AWS Infraestructure

### **Puntos clave:**

1. **Infraestructura global de AWS**:
   * Está compuesta por **regiones** y **zonas de disponibilidad**.
   * Actualmente hay **31 regiones** y **99 zonas de disponibilidad** (estos números cambian con el tiempo).
2. **Región**:
   * Es una **ubicación geográfica** como Londres, Sídney o Virginia del Norte.
   * Cada región tiene **dos o más zonas de disponibilidad**.
3. **Zona de disponibilidad**:
   * Es un grupo de **centros de datos** que están cerca unos de otros (dentro de un radio de 100 km o 60 millas).
   * Puede haber varios centros de datos en una zona de disponibilidad.
4. **Edge Locations**:
   * Son **puntos finales** que AWS utiliza para **almacenar en caché contenido**.
   * Por ejemplo, **CloudFront**, la red de entrega de contenido de Amazon, usa estas ubicaciones.
5. **AWS Management Console**:
   * La consola de administración de AWS ha evolucionado a lo largo del tiempo y ahora incluye muchos más servicios que antes.
   * Algunos servicios importantes para el examen de **Architect Associate** incluyen **Compute (EC2, Lambda)**, **almacenamiento (S3, Glacier)** y **bases de datos**.
6. **Consejos para el examen**:
   * Solo necesitas conocer **los servicios principales** para aprobar el examen, no todos los servicios de AWS.

Una **región** es una ubicación geográfica con varias zonas de disponibilidad, una **zona de disponibilidad** es un conjunto de centros de datos, y las **ubicaciones de borde** son utilizadas para la distribución de contenido mediante servicios como **CloudFront**

# Shared Responsibility Model

**Responsabilidad de AWS (seguridad de la nube)**:

* AWS es responsable de la **seguridad de la infraestructura** en la nube. Esto incluye:
  + Las regiones y zonas de disponibilidad.
  + La seguridad física de los centros de datos (vallas, seguridad, etc.).
  + El hardware, almacenamiento, red y la capa de virtualización (hipervisor).
  + El software que ejecuta los servicios en la nube.

**Responsabilidad del usuario (seguridad dentro de la nube)**:

* Como usuario, eres responsable de la **seguridad dentro de la nube**. Esto incluye:
  + La gestión de los **datos de los clientes**.
  + La configuración de los **sistemas operativos** en las máquinas virtuales (EC2, por ejemplo).
  + El **manejo de aplicaciones** y las configuraciones de red y firewalls.
  + La administración de **Identidad y Gestión de Acceso (IAM)**.
  + El **cifrado de datos** tanto en el lado del cliente como en el servidor.
  + Proteger el tráfico de red.

**Responsabilidad compartida**:

* Un área clave donde la responsabilidad es **compartida** es en el **cifrado**:
  + El cliente puede habilitar el cifrado de los volúmenes o datos.
  + AWS debe asegurarse de que esos datos efectivamente estén cifrados correctamente una vez que el cliente lo configure.

**Consejo para el examen**:

* En los exámenes de AWS, se plantean escenarios de responsabilidad compartida. Un consejo clave es preguntarte si puedes controlar algo desde la **consola de administración de AWS**:
  + Si puedes, entonces es tu **responsabilidad**.
  + Si no puedes controlarlo (como la infraestructura física), entonces es responsabilidad de **AWS**.

# Compute, Storage, Databases and Networking

1. **Introducción a los cuatro componentes clave del examen**:
   * Para facilitar el estudio del examen AWS Certified Solutions Architect - Associate, se dividen los servicios en cuatro componentes principales:
     + **Cómputo**
     + **Almacenamiento**
     + **Bases de datos**
     + **Redes**
   * Tener un buen entendimiento de estos cuatro temas es clave para aprobar el examen.
2. **Cómputo**:
   * Se refiere al poder de procesamiento que se utiliza para ejecutar aplicaciones, como servidores o máquinas virtuales.
   * Los servicios importantes que se cubren son:
     + **EC2**: Máquinas virtuales.
     + **Lambda**: Computación sin servidor (solo se gestiona el código).
     + **Elastic Beanstalk**: Automatiza la implementación de aplicaciones en AWS.
3. **Almacenamiento**:
   * Es el lugar donde se guarda la información de manera segura en la nube.
   * Los servicios clave incluyen:
     + **S3**: Servicio de almacenamiento simple.
     + **EBS**: Almacenamiento en bloque para máquinas virtuales.
     + **EFS**: Servicio de archivos elásticos.
     + **FSx** y **Storage Gateway**.
4. **Bases de datos**:
   * Almacenan y recuperan datos de manera confiable.
   * Los servicios que se cubrirán son:
     + **RDS**: Servicio de bases de datos relacionales.
     + **DynamoDB**: Base de datos no relacional.
     + **Redshift**: Tecnología de almacenamiento de datos (data warehousing).
5. **Redes**:
   * Permite que los componentes (cómputo, almacenamiento, bases de datos) se comuniquen entre sí.
   * Los servicios clave son:
     + **VPC**: Centros de datos virtuales en la nube.
     + **Direct Connect**: Conexión directa entre centros de datos locales y AWS.
     + **Route 53**: Sistema de nombres de dominio (DNS).
     + **API Gateway**: Forma sin servidor de gestionar servidores web.
     + **AWS Global Accelerator**: Acelera el acceso a las aplicaciones.
6. **Guía del examen**:
   * El examen tiene 65 preguntas, de las cuales solo 50 cuentan para la puntuación final.
   * Se dividen en cuatro dominios:
     + Diseñar arquitecturas seguras.
     + Diseñar arquitecturas resilientes.
     + Diseñar arquitecturas de alto rendimiento.
     + Diseñar arquitecturas optimizadas en costos.
   * La guía del examen incluye una lista de servicios en el alcance del examen, que se cubren en diferentes niveles de profundidad según su importancia.
7. **Servicios clave para el examen**:
   * **Cómputo**: EC2, Lambda, Elastic Beanstalk.
   * **Almacenamiento**: S3, EBS, EFS, FSx, Storage Gateway.
   * **Bases de datos**: RDS, DynamoDB, Redshift.
   * **Redes**: VPC, Direct Connect, Route 53, API Gateway, AWS Global Accelerator.

El curso comienza cubriendo la **gestión de identidad y acceso (IAM)**, que es esencial para crear usuarios y grupos, y luego pasa al tema de **almacenamiento**, empezando por S3.

# Well-Architected Framework

Vamos a hablar brevemente sobre el **Well-Architected Framework**. Primero, mencionamos los whitepapers, que se pueden obtener en el sitio **aws.amazon.com/whitepapers**. No necesitas leer todos los whitepapers para aprobar el examen, pero el más importante es el **Resumen del Marco Well-Architected de AWS**, que consta de seis pilares. Es recomendable leer este whitepaper al final del curso, antes del examen, ya que el examen está estructurado en torno a este marco.

Los seis pilares del Marco Well-Architected son:

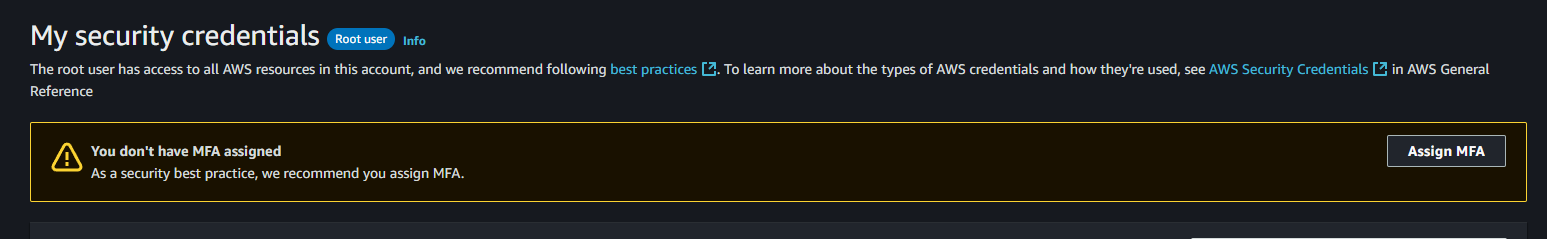
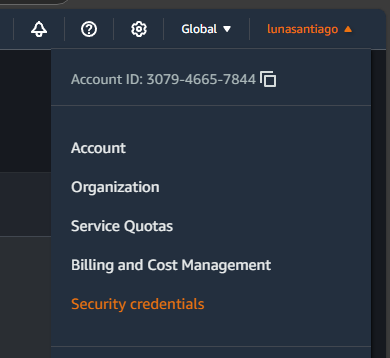
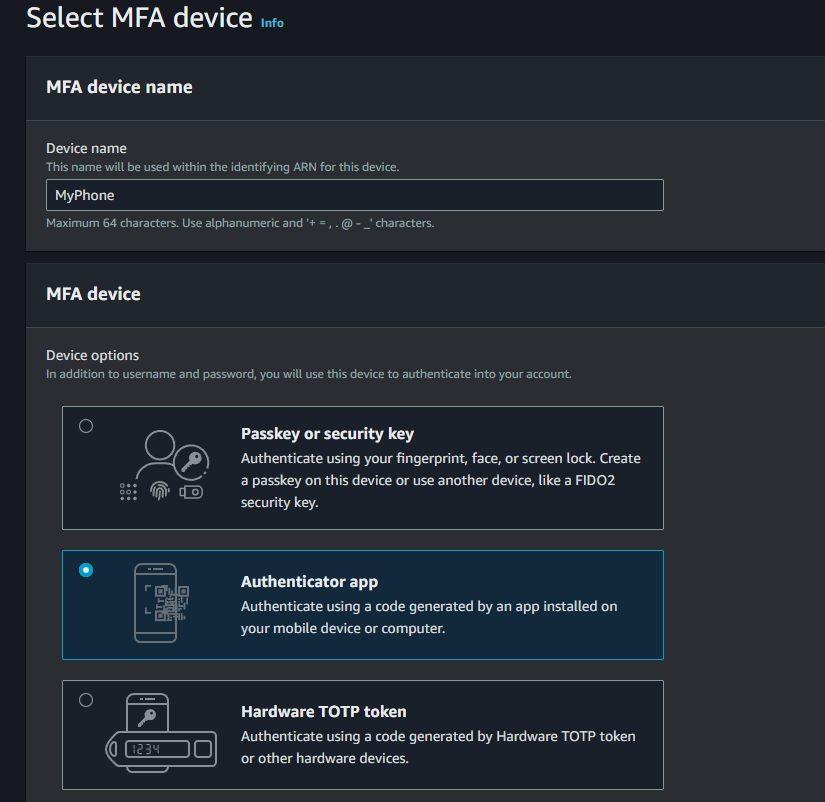
1. **Excelencia operativa**: Ejecutar y monitorear sistemas para entregar valor y mejorar continuamente los procesos.
2. **Eficiencia en el rendimiento**: Utilizar los recursos de TI de manera eficiente.
3. **Seguridad**: Proteger la información y los sistemas.
4. **Optimización de costos**: Evitar costos innecesarios.
5. **Confiabilidad**: Garantizar que una carga de trabajo funcione correctamente y de manera consistente.
6. **Sostenibilidad**: Minimizar el impacto ambiental de las cargas de trabajo en la nube.

Es recomendable revisar el whitepaper justo antes del examen, ya que contiene muchos detalles técnicos que aprenderás a lo largo del curso

# Identity and Access Management (IAM)

permite gestionar usuarios y sus niveles de acceso a los recursos de AWS. Se mencionan los siguientes puntos:

1. **IAM (Identity Access Management)**: Es una herramienta para gestionar quién puede acceder a qué en AWS. Permite crear usuarios, roles y grupos, y asignarles permisos específicos. Es esencial entender IAM para administrar AWS de manera efectiva y para aprobar los exámenes relacionados con AWS.
2. **Cuenta raíz**: La cuenta raíz es la dirección de correo electrónico con la que se creó la cuenta de AWS y tiene acceso total a todos los servicios de AWS. Es crucial proteger esta cuenta para evitar que alguien acceda y use los recursos sin control.
3. **Seguridad de la cuenta raíz**: La mejor forma de proteger la cuenta raíz es habilitar la autenticación multifactor (MFA). En la consola de AWS, se puede asignar un dispositivo MFA utilizando aplicaciones como Google Authenticator o Authy. Estas aplicaciones generan códigos de seguridad temporales que se deben ingresar junto con la contraseña.
4. **Configurar MFA**: Para activar MFA, se escanea un código QR desde una aplicación autenticadora en el teléfono. Luego, se ingresan los códigos generados para finalizar la configuración. Esto agrega una capa adicional de seguridad, evitando que alguien acceda solo con la contraseña.
5. **Recomendaciones para el examen**: Siempre se debe habilitar MFA en la cuenta raíz. También se recomienda crear un grupo de administradores con los permisos adecuados y asignarles usuarios para evitar usar directamente la cuenta raíz. Próximamente, se abordarán los permisos y cómo se pueden asignar a grupos, usuarios y roles en IAM.

Asignar MFA:  
  


## 

## IAM Policy Documents

Cómo controlar las acciones de los usuarios en AWS mediante **documentos de políticas IAM**. Los puntos principales son:

1. **Permisos en IAM**: Se asignan permisos a través de documentos de políticas escritos en formato **JSON** (JavaScript Object Notation). Estos documentos definen acciones permitidas o denegadas para los usuarios, roles o grupos.
2. **Estructura de un documento de política**: Un documento de política contiene declaraciones con elementos clave como "Effect" (efecto), "Action" (acción) y "Resource" (recurso). Por ejemplo, un documento que otorga acceso administrativo completo tendría "Effect": "Allow" y utilizaría comodines (\*) para permitir todas las acciones en todos los recursos.
3. **Mejores prácticas**: No es recomendable asignar directamente las políticas a usuarios individuales, ya que es difícil de gestionar. Es mejor crear grupos, asignar las políticas a esos grupos y luego añadir a los usuarios al grupo para que hereden los permisos.
4. **IAM a nivel global**: IAM funciona a nivel global, lo que significa que un usuario o política creada está disponible en todas las regiones de AWS.
5. **Uso de políticas gestionadas por AWS**: AWS ofrece más de 1000 políticas predefinidas que pueden usarse para tareas comunes, como "AdministratorAccess", que otorga acceso completo a todos los servicios de AWS.
6. **Consejos para el examen**: Es importante estar familiarizado con la lectura de documentos JSON de políticas IAM. Se recomienda explorar diferentes políticas en la consola de AWS para entender cómo se asignan los permisos.

**Bloques fundamentales de IAM**:

* **Usuarios**: Deben ser personas físicas, y cada usuario representa a una sola persona.
* **Grupos**: Los usuarios se agrupan según su función laboral (administradores, desarrolladores, finanzas, etc.) para gestionar permisos de manera más eficiente.
* **Roles**: Permiten que una parte de AWS acceda a otra, como una máquina virtual accediendo a un bucket S3.

**Mejores prácticas**:

* Es mejor asignar permisos a través de grupos, no a usuarios individuales, ya que es más fácil de gestionar.
* No se deben compartir cuentas de usuario entre varias personas.
* Seguir el **principio del mínimo privilegio**, asignando solo los permisos necesarios para realizar el trabajo.

**Configuración en AWS Console**:

* Se mostró cómo crear un usuario, asignarlo a un grupo y añadir permisos mediante políticas predefinidas por AWS.
* Los usuarios sin grupo o política asignada solo pueden cambiar su contraseña.

**Acceso programático**:

* Se explicó cómo generar claves de acceso para usar el CLI de AWS, separadas de las credenciales de inicio de sesión.

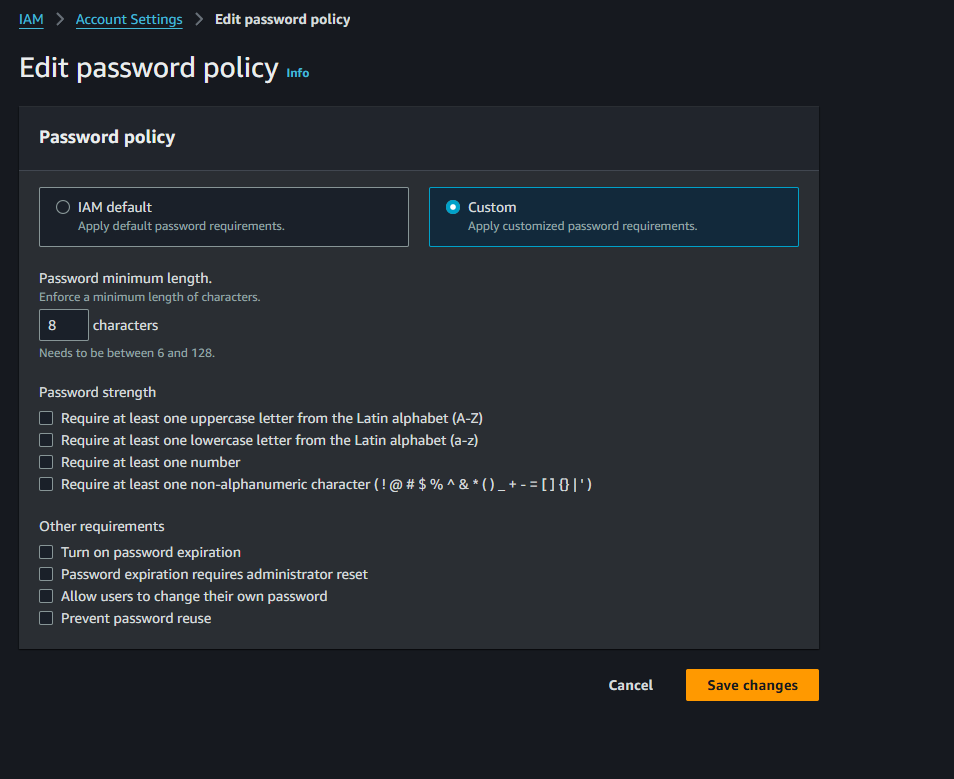
**Proveedores de identidad**:

* IAM permite integrar con servicios como Active Directory o Google para utilizar Single Sign-On (SSO).

**Consejos para el examen**:

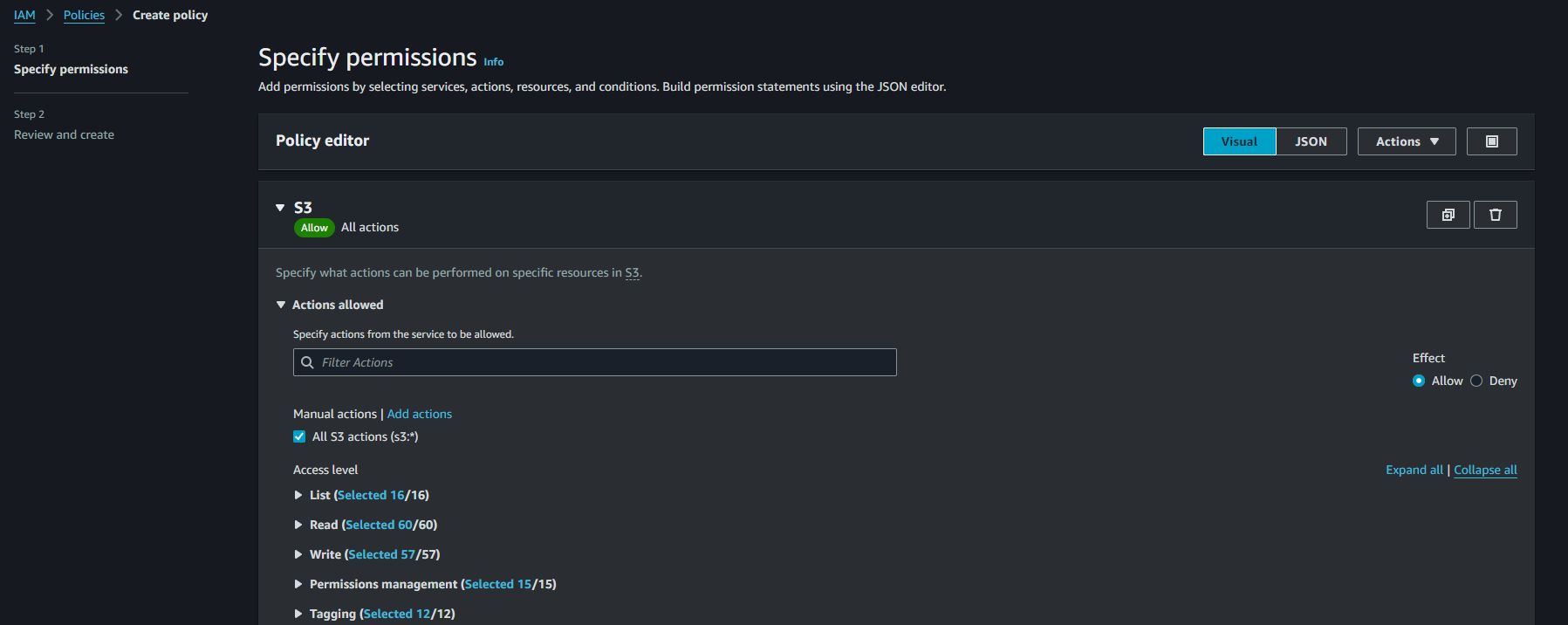
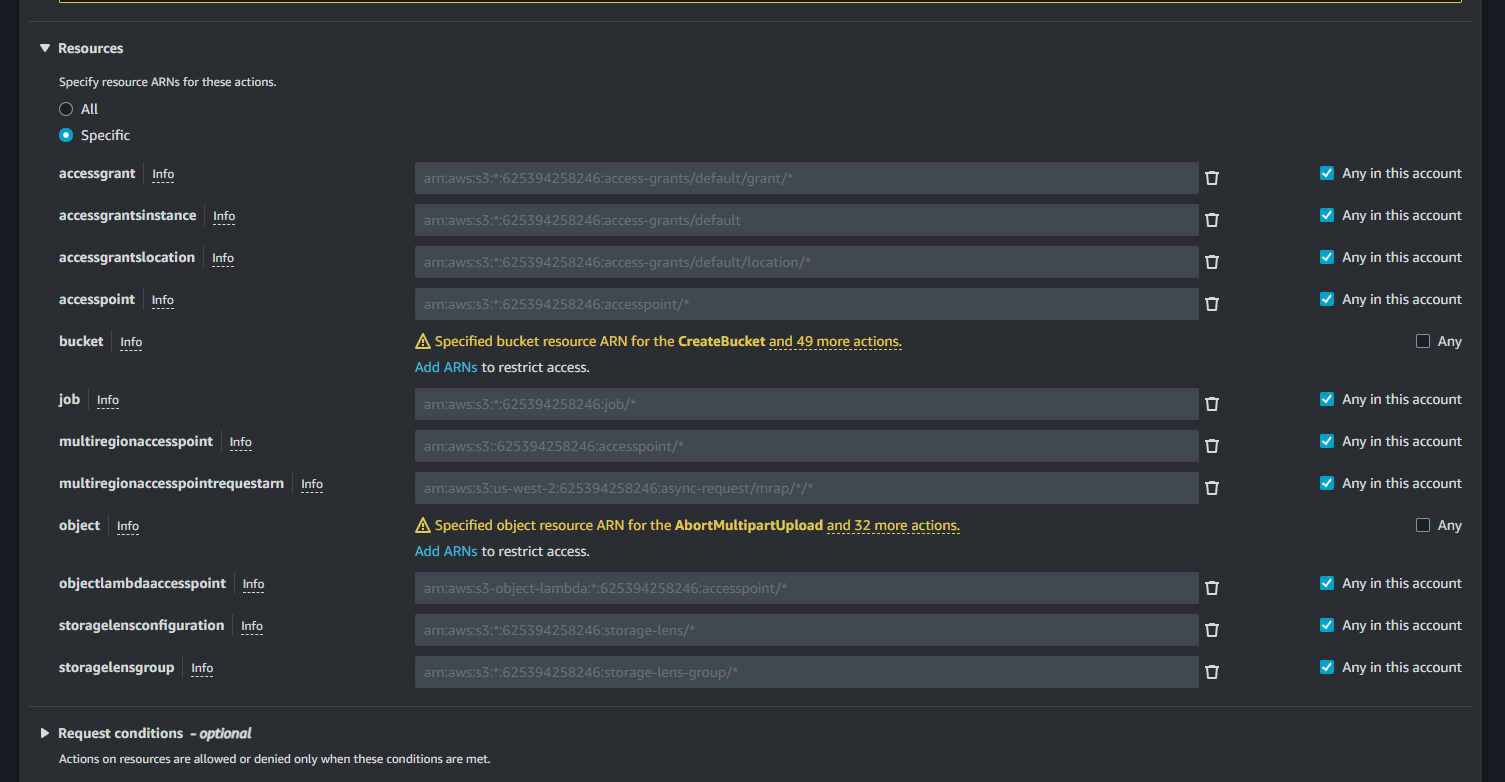
* IAM es global, no está limitado a regiones.
* La cuenta raíz tiene acceso completo y debe protegerse.
* Los usuarios recién creados no tienen permisos hasta que se les asignen.
* IAM permite la federación de identidades, integrando credenciales de Active Directory con AWS mediante SAML.

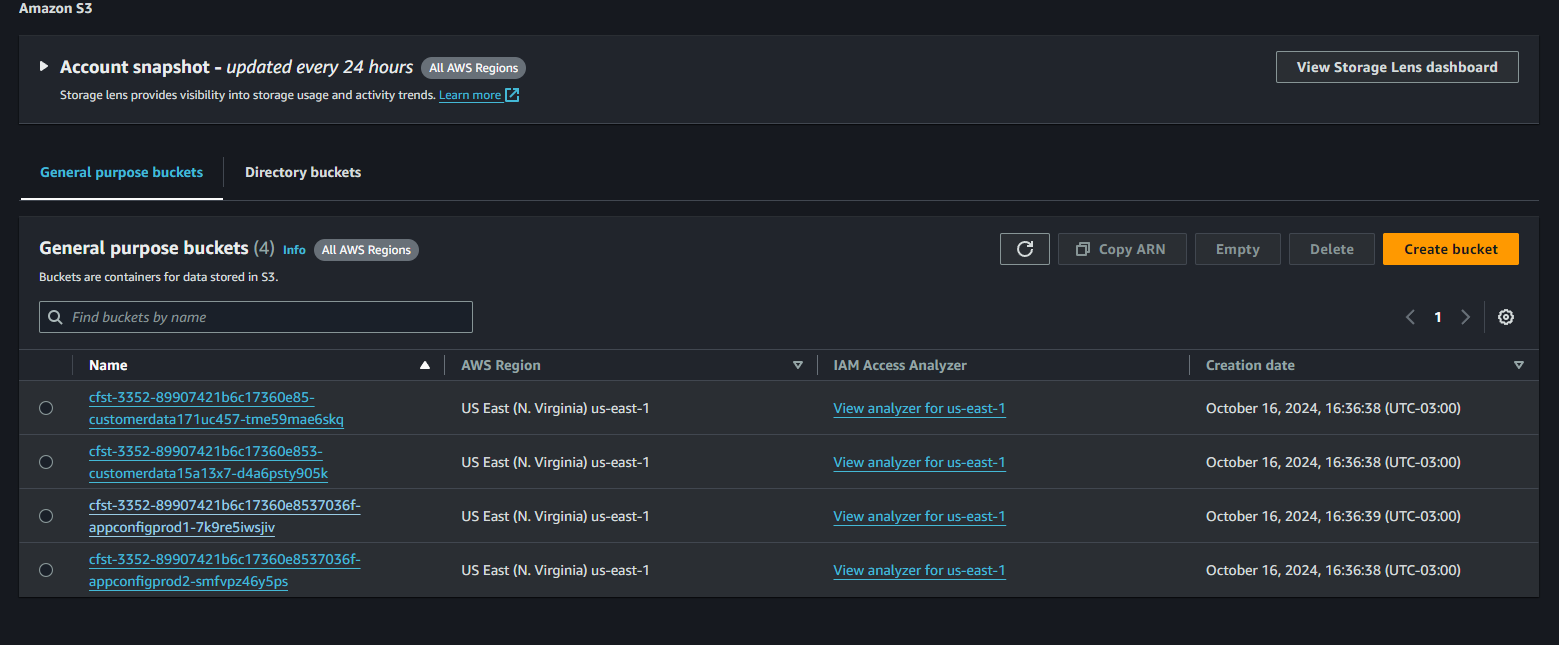
Politicas de Pass



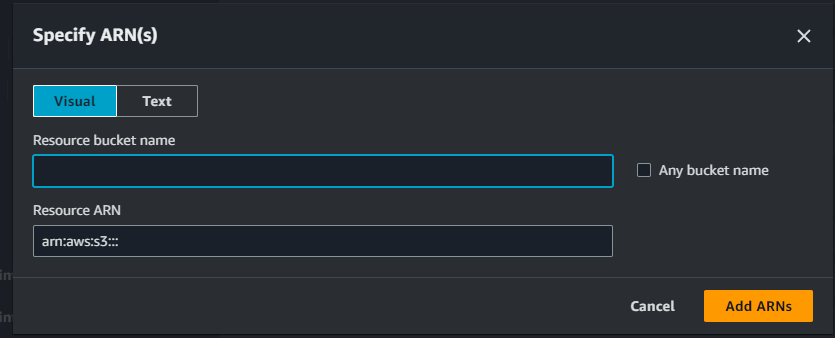
## 

Crear Policies: en este caso vamos a crear una policy llamada **S3RestrictedPolicy** que contenga acceso a determinados buckets

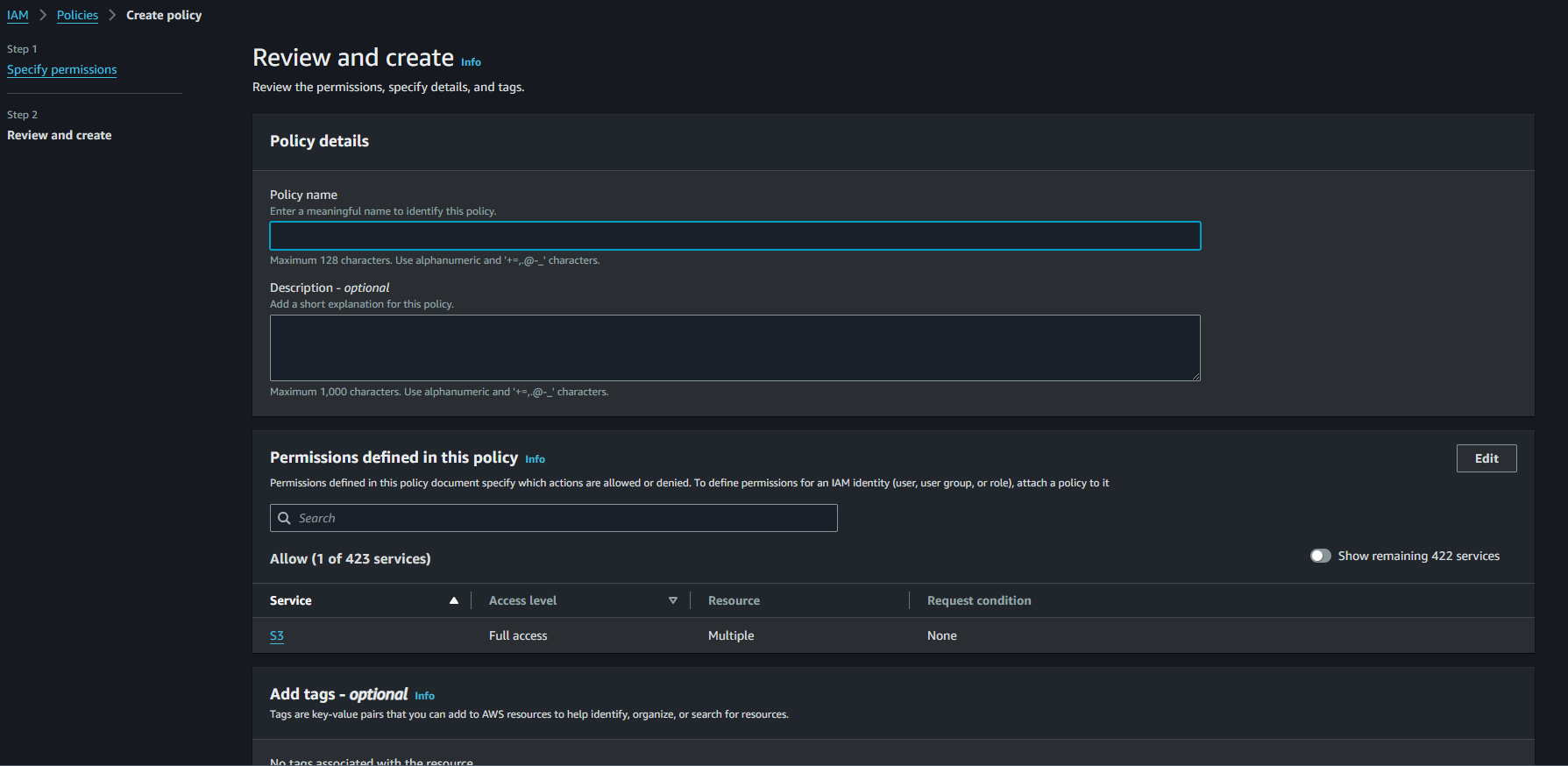
Seleccionamos el servicio S3Seleccionamos todos los recursos excepto bucket ya que le enviaremos los buckets especificos En otra pestana abrimos S3 copiaremos el nombre de los buckets appconfig



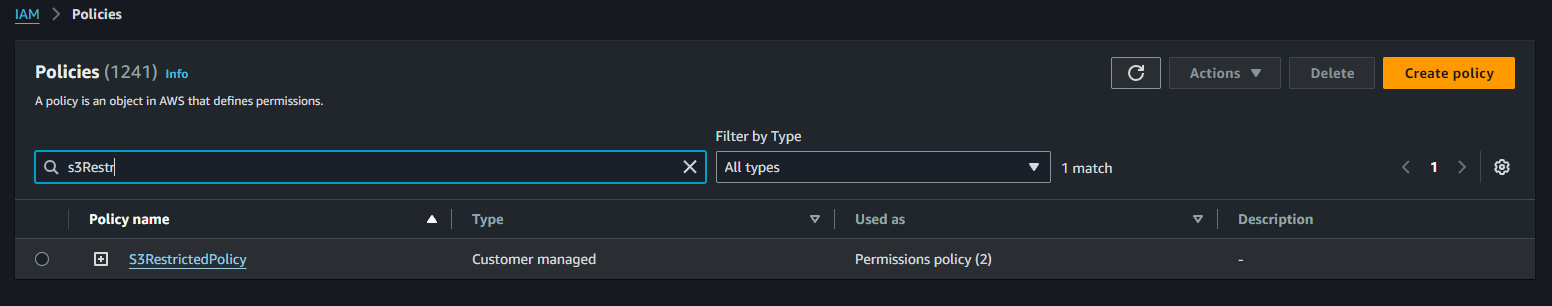
Volvemos a la policy, click en add arn y pegaremos el nombre de los buckets copiados

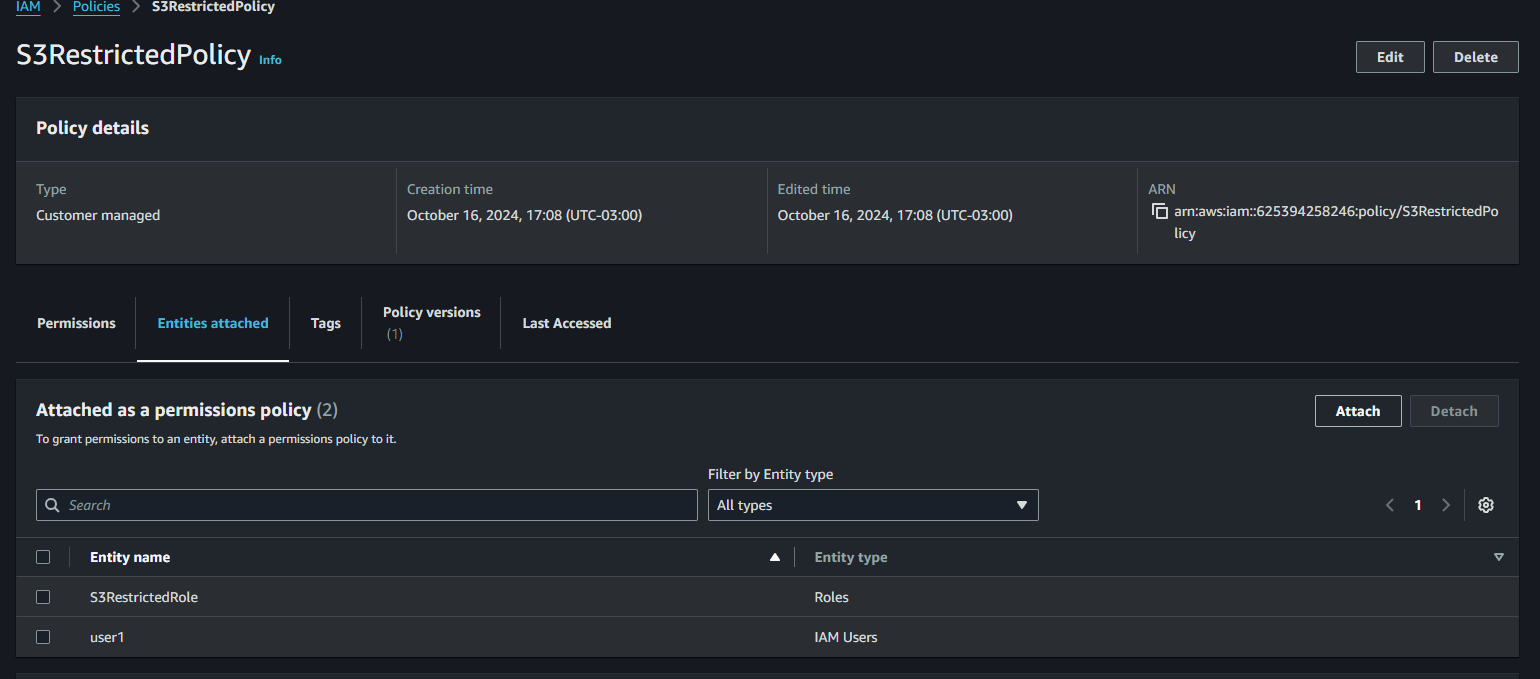


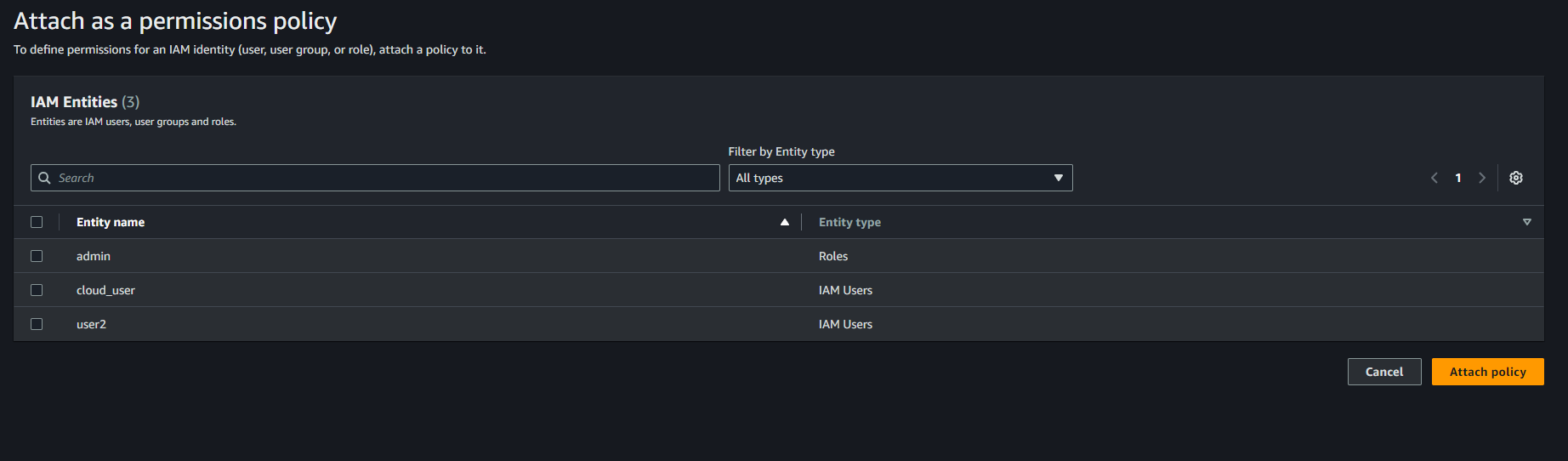
Click en Add ARN, luego en next y veremos la siguiente ventana donde colocaremos el nombre de la politica

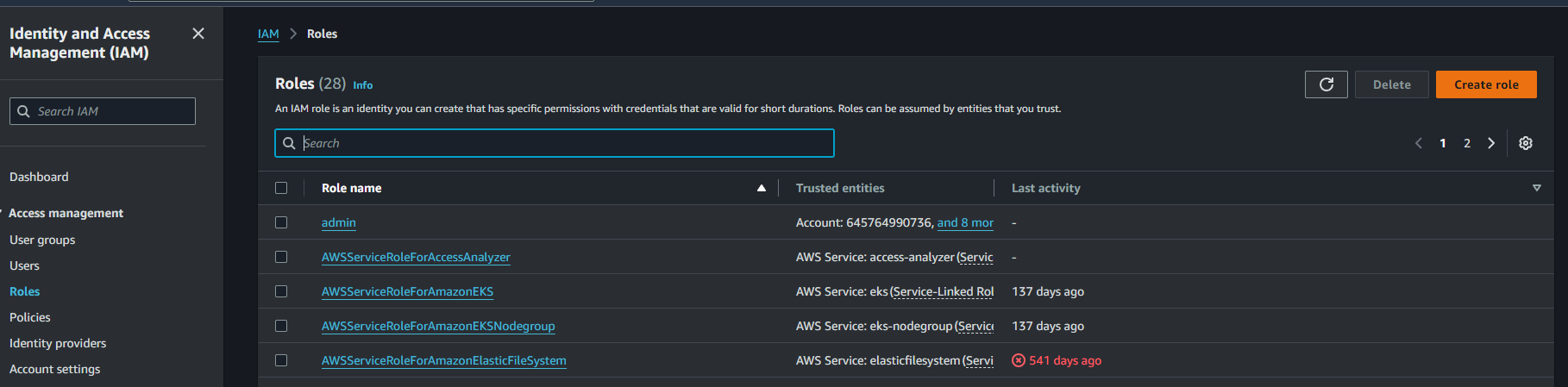


Despues veremos la policy

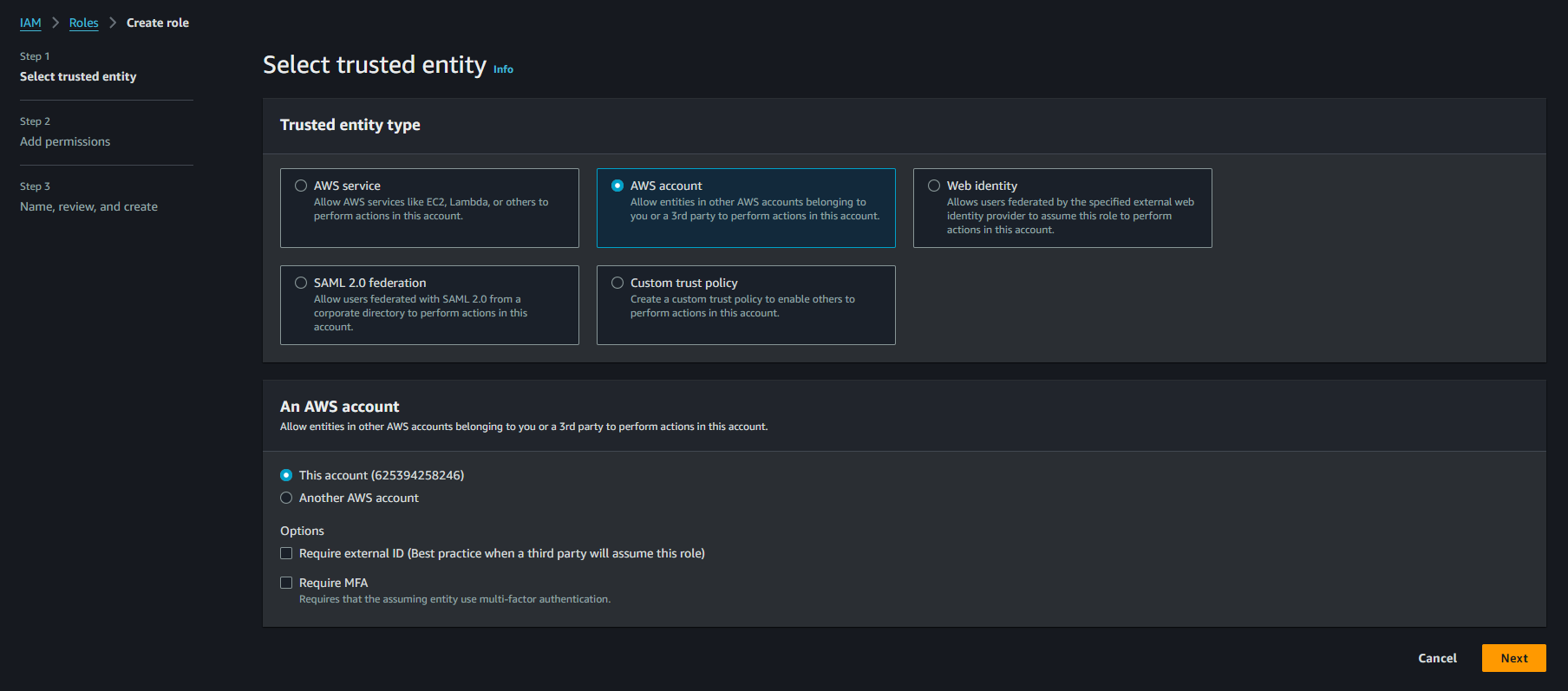
Entramos en ella e iremos a la parte de entities attached donde agregaremos a un user

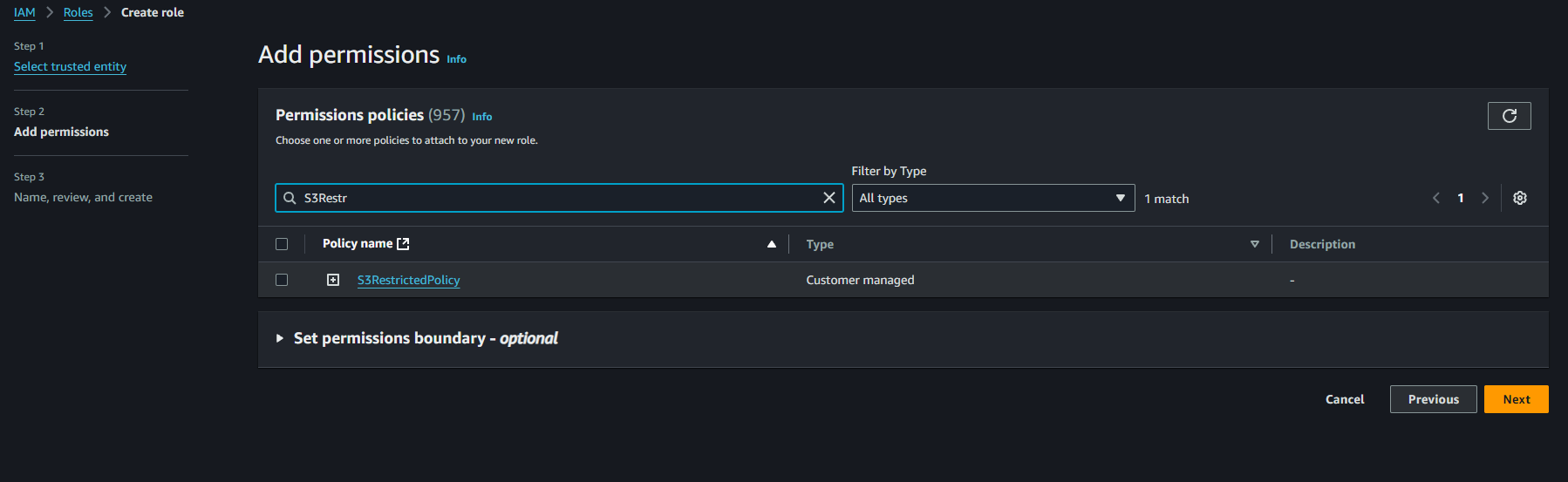
Seleccionamos un usuario y attach policy

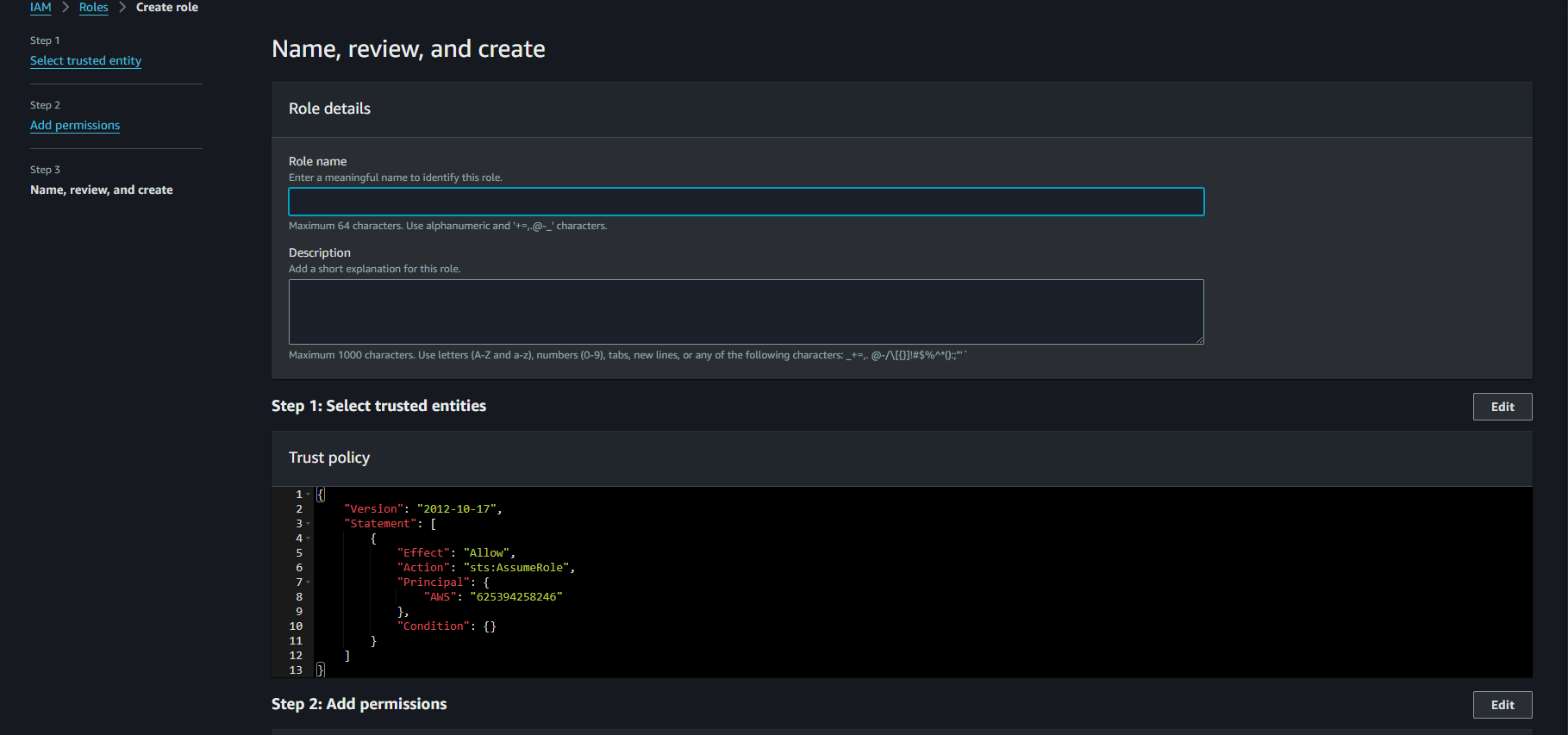
Ahora se procede con la creacion de un rol que contenga dichas policies. Vamos a IAM, Role y Create Role.



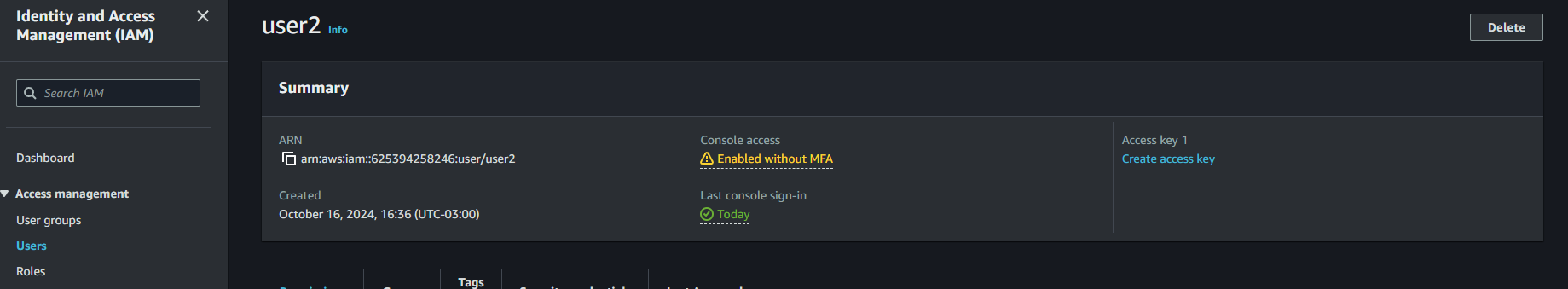
Seleccionamos AWS account y next

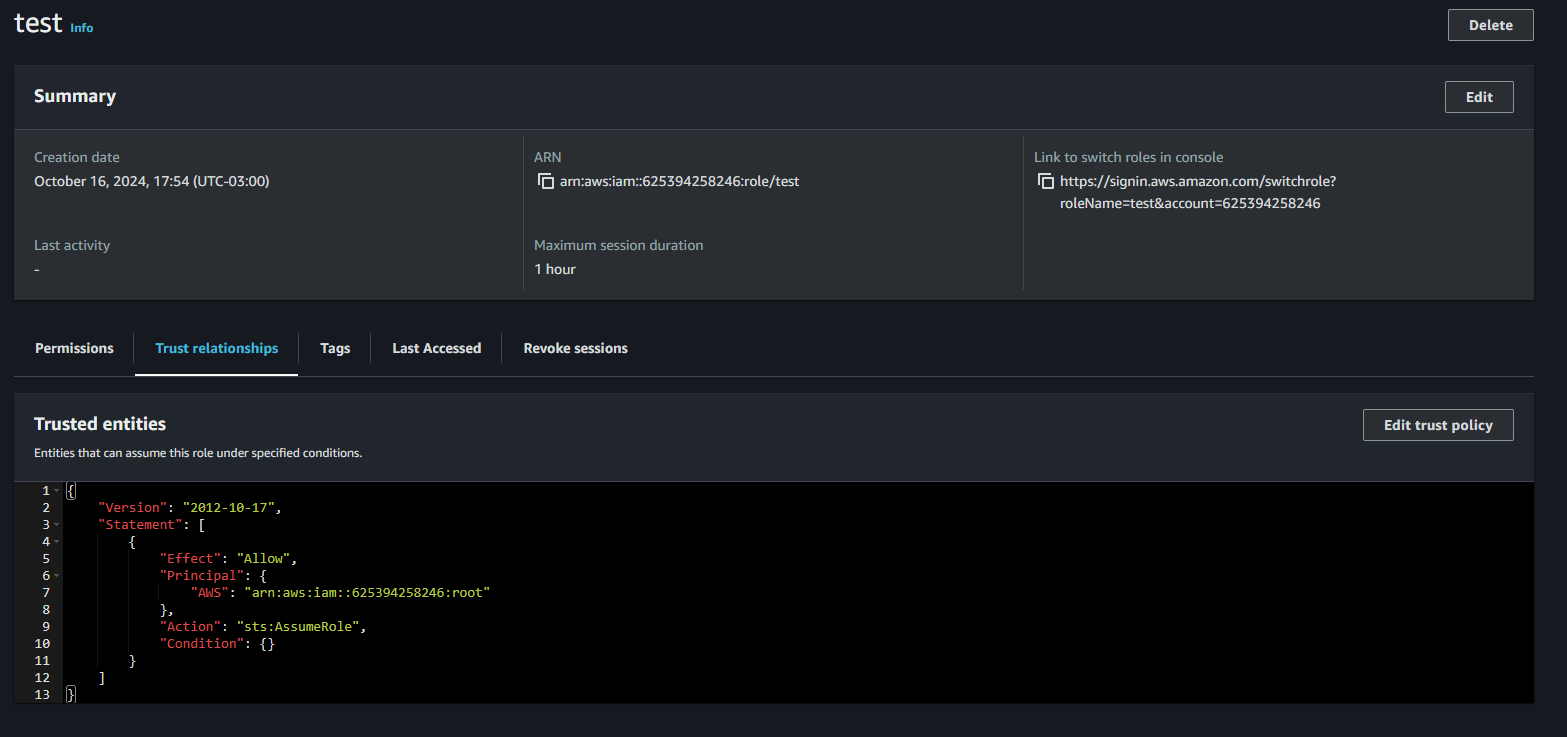
Dentro de las Policies, elegimos la que creamos anteriormente

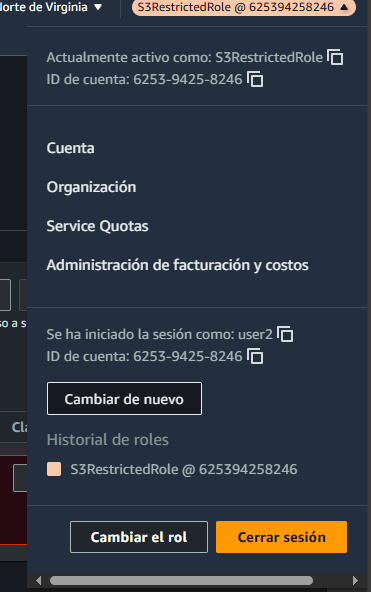
Elegimos un nombre para el rol y luego create role



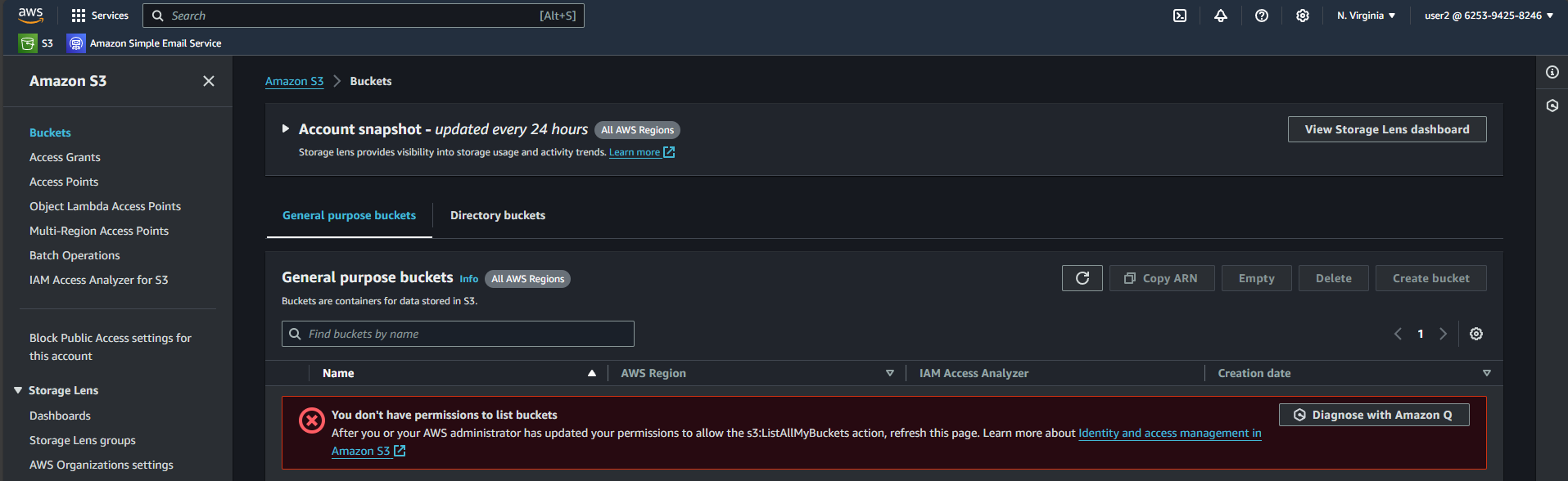
Ahora hay que asegurarnos que el usuario2 tenga acceso para asumir dicho rol creado. Para esto iremos a Users, seleccionamos el usuario2 y copiamos su arn

Volvemos a Roles, entramos en nuestro rol creado e iremos a trust relationships, edit trust, veremos un codigo y dentro de el buscaremos arn y colocamos el arn copiado del usuario2

De esta manera el usuario2 tendra un rol extra con el cual puede ver los buckets, caso contrario no podra visualizarlos y cada vez que quiera cambiar de rol debera colocar switch role y para volver al anterior switch back



User2 sin rol para ver buckets

User2 con rol para ver buckets