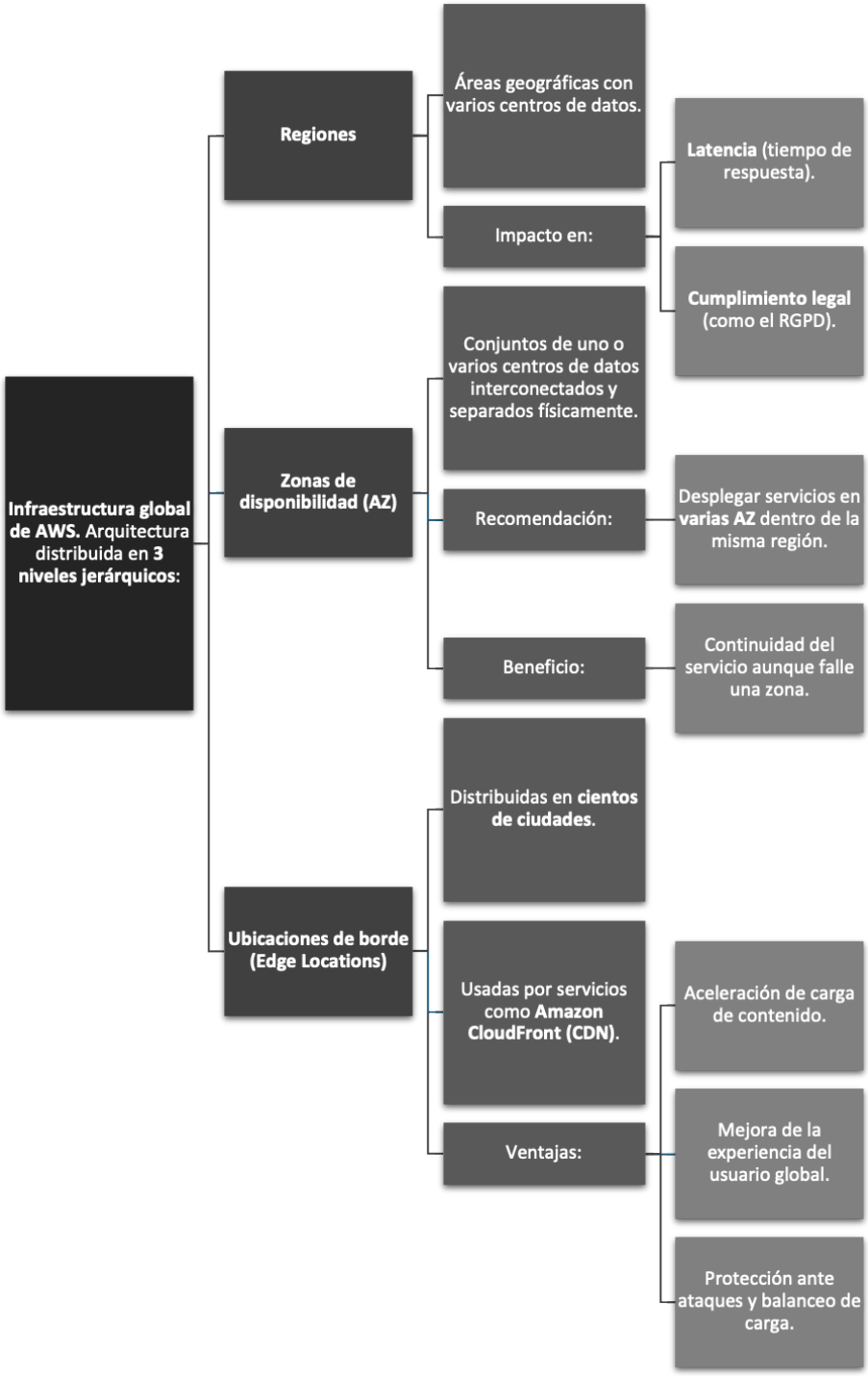
3

IMPLEMENTACIÓN DE SERVICIOS BÁSICOS EN LA NUBE

Con la cuenta creada y el entorno bajo control, llega lo interesante: empezar a desplegar servicios reales. En este bloque se trabajará directamente con los recursos más comunes y útiles dentro de AWS. Se aprenderá cómo crear instancias de computación (EC2), cómo almacenar información en Amazon S3, cómo alojar una página web estática, cómo distribuir la carga entre servidores o cómo aplicar políticas de control de acceso con IAM. También se verán soluciones de almacenamiento como EFS o EBS y se introducirá el concepto de soporte técnico para resolver incidencias. La idea es que el alumno o la alumna tenga una primera experiencia funcional con la nube, construyendo pequeños proyectos que imiten lo que luego puede encontrarse en entornos profesionales. Es el momento de ensuciarse las manos, pero siempre con guías claras y explicaciones paso a paso.

3.1 LA NUBE DE AWS

Imagina que estás montando una empresa que va a operar a nivel mundial. Necesitas servidores rápidos, estables y seguros repartidos por todo el planeta. ¿Cómo te aseguras de que un cliente en Madrid, otro en Tokio y otro en Buenos Aires tengan la misma experiencia fluida y sin retrasos al usar tu plataforma? Aquí es donde la infraestructura global de Amazon Web Services (AWS) entra en juego. AWS no es solo una colección de servicios sueltos conectados entre sí: detrás hay una arquitectura sólida, bien pensada y desplegada por todo el mundo para ofrecer velocidad, disponibilidad y fiabilidad a escala global.


Infraestructura global de AWS. Arquitectura distribuida en 3 niveles jerárquicos:
Regiones
Áreas geográficas con varios centros de datos.
Impacto en:
Latencia (tiempo de respuesta).
Cumplimiento legal (como el RGPD).
Zonas de disponibilidad (AZ)
Conjuntos de uno o varios centros de datos interconectados y separados físicamente.
Recomendación:
Desplegar servicios en varias AZ dentro de la misma región.
Beneficio:
Continuidad del servicio aunque falle una zona.
Ubicaciones de borde (Edge Locations)
Distribuidas en cientos de ciudades.
Usadas por servicios como Amazon CloudFront (CDN).
Ventajas:
Aceleración de carga de contenido.
Mejora de la experiencia del usuario global.
Protección ante ataques y balanceo de carga.

AWS ha creado una red distribuida en tres niveles principales: regiones, zonas de disponibilidad (availability zones) y ubicaciones de borde (edge locations). Estos tres conceptos forman la columna vertebral de todo lo que funciona en la nube de Amazon. Entender esta estructura ayuda a comprender por qué muchos servicios pueden escalar tan bien, ser tan rápidos y recuperarse con tanta eficacia ante fallos o picos de demanda.

Las regiones son el nivel más alto de esta jerarquía. Una región es un área geográfica que agrupa varios centros de datos. Por ejemplo, AWS tiene regiones en ciudades como París, Fráncfort, São Paulo o Tokio. Y sí, también hay una en España: la región Europa (España), ubicada en Aragón, se ha convertido en una de las apuestas más fuertes para el mercado hispanohablante. Cada región es independiente y está diseñada para garantizar aislamiento a nivel de datos, lo que es muy importante para cumplir con regulaciones locales como el RGPD. Cuando eliges una región para desplegar tus servicios, estás decidiendo dónde estarán físicamente tus recursos. Esto tiene impacto tanto en la latencia (el tiempo de respuesta del sistema) como en el cumplimiento legal, así que no es una elección aleatoria.

Dentro de cada región, AWS divide su infraestructura en zonas de disponibilidad. Una zona de disponibilidad no es otra cosa que uno o varios centros de datos independientes entre sí pero interconectados por redes de alta velocidad y baja latencia. Están físicamente separadas para que, si ocurre un fallo en una zona —por ejemplo, un apagón o una avería—, las otras zonas de la misma región

sigan funcionando sin problema. Este diseño permite que las aplicaciones sean más resistentes y puedan recuperarse rápidamente si algo falla. De hecho, una buena práctica en AWS es desplegar tus servicios en múltiples zonas de disponibilidad dentro de una misma región. Así, si se cae una parte, el resto sigue funcionando y tus usuarios ni se enteran.

Y aún hay otro nivel que completa el mapa: las edge locations, o ubicaciones de borde. Aquí es donde entra el mundo del contenido web, la aceleración de cargas y la distribución eficiente. Las edge locations están repartidas en cientos de ciudades y funcionan como puntos de entrada o de distribución de datos. Son clave cuando se usan servicios como Amazon CloudFront, que es la red de entrega de contenido (CDN) de AWS. Gracias a estas ubicaciones de borde, si alguien entra en tu web desde Lima, los archivos estáticos (como imágenes o vídeos) no tienen que viajar desde Irlanda, sino desde un nodo cercano en Sudamérica. Esto mejora muchísimo la velocidad de carga y la experiencia del usuario.

AWS no funciona como una gran nube centralizada que lo gestiona todo desde un único lugar. Funciona como una red bien orquestada de regiones, zonas y nodos que permite a las empresas desplegar sus soluciones donde más sentido tenga para su negocio, sus usuarios y su contexto legal. Puedes pensar en ello como una especie de sistema nervioso planetario: cada región es como un órgano principal, cada zona de disponibilidad una parte del mismo que actúa con autonomía y cada edge location un punto de contacto rápido para el entorno. Esta forma de organizar los recursos es la que hace posible que miles de empresas, desde startups hasta multinacionales, funcionen sin interrupciones, con buen rendimiento y con capacidad para crecer sin límites.

Entender esta arquitectura global es una herramienta para anticiparse a posibles problemas y optimizar el rendimiento sin complicarse la vida. Por ejemplo, si estás desarrollando una aplicación que tendrá usuarios en Europa y América Latina, puedes desplegar tus servicios principales en la región de España o en la de Brasil, y usar edge locations cercanas a tus usuarios finales para que todo cargue más rápido, sin necesidad de invertir en servidores físicos en cada país. Este enfoque también permite experimentar con configuraciones multirregión, en las que puedes tener servicios replicados en distintos puntos del planeta, lo cual es muy útil si tu proyecto empieza a crecer y necesitas dar un salto de escala sin perder calidad ni estabilidad.

Otra ventaja que ofrece esta estructura es la flexibilidad para adaptarte a contextos específicos. En el caso de España, por ejemplo, muchas organizaciones públicas y privadas se ven obligadas a cumplir con regulaciones estrictas sobre dónde pueden estar los datos. Gracias a la región en Aragón, estas entidades pueden usar servicios avanzados de AWS sin salirse del marco legal. Y lo mismo pasa con empresas que necesitan cumplir con normativas del sector salud, financiero o educativo. Elegir bien la región y conocer el mapa completo de la infraestructura global de AWS te permite diseñar soluciones alineadas con las exigencias del entorno, sin tener que hacer malabares técnicos o renunciar a determinadas herramientas.

Además, las zonas de disponibilidad dentro de una región están pensadas para que puedas distribuir tu aplicación de forma inteligente. Supón que montas una base de datos que no puede caerse bajo ningún concepto. Si la alojas en una sola zona y esa zona tiene un problema (por mantenimiento o fallo inesperado), tu aplicación se vería afectada. Pero si replicaste esa base de datos en al menos dos zonas distintas de la misma región, tu sistema seguirá operativo y nadie lo notará. Este tipo de configuración se llama alta disponibilidad, y es una de las razones por las que AWS ha ganado tanta popularidad entre desarrolladores y empresas que necesitan fiabilidad en todo momento.

Respecto a las edge locations, no solo sirven para entregar contenido más rápido. También tienen un papel muy importante en temas de seguridad, balanceo de carga y autenticación. Muchos servicios de AWS se apoyan en estas ubicaciones de borde para interceptar ataques, bloquear tráfico sospechoso o validar accesos sin que la solicitud tenga que viajar hasta el centro de datos principal. Esto significa que tus sistemas pueden responder más rápido a amenazas y ofrecer una experiencia fluida incluso cuando hay picos de tráfico o intentos de saturación. Es como tener pequeñas torres de vigilancia repartidas por todo el mundo, protegiendo tu castillo digital desde fuera.

Nota

Hablar de la infraestructura global de AWS no es hablar de una red abstracta o de cables invisibles flotando en el aire. Es hablar de decisiones reales que impactan directamente en el día a día de cualquier proyecto que se desarrolle en la nube. Saber cómo se distribuyen las regiones, cómo funcionan las zonas de disponibilidad y qué papel juegan las edge locations es como tener el plano de una ciudad antes de construir un edificio: te permite elegir el mejor terreno, prever los riesgos, aprovechar mejor los recursos y ofrecer a tus usuarios una experiencia sólida, rápida y segura desde cualquier parte del mundo.

3.2 AWS GESTIÓN DE IDENTIDAD Y ACCESO (IAM)

Cuando se empieza a trabajar con Amazon Web Services (AWS), una de las primeras cosas que hay que aprender es cómo controlar quién puede hacer qué dentro de la cuenta. Y aquí entra en juego un servicio fundamental: IAM, o lo que es lo mismo, **Identity and Access Management**.

IAM es el servicio que te permite **crear y administrar usuarios, roles y permisos** dentro de tu entorno de AWS. En otras palabras, es el guardián de tu nube:

quien decide qué puede hacer cada persona, aplicación o servicio que interactúe con tu infraestructura. Cuando creas tu cuenta de AWS por primera vez, se genera automáticamente un **usuario root**, que es como el súper administrador: tiene acceso total y sin restricciones. Y aquí viene el primer consejo importante: **ese usuario hay que guardarlo bajo llave y usarlo lo menos posible**. Lo recomendable es crear enseguida usuarios individuales con los permisos necesarios para cada tarea y desactivar el uso cotidiano del root, porque un descuido con ese usuario puede afectar a toda la cuenta.

Una vez tienes claro eso, el siguiente paso es crear **usuarios de IAM**, que pueden ser personas (por ejemplo, miembros de tu equipo de desarrollo) o aplicaciones que necesitan interactuar con tus recursos (como un sistema automático que lanza backups todas las noches). A cada usuario se le asignan unas credenciales, que pueden incluir nombre, contraseña, claves de acceso y MFA (autenticación multifactor). Esto último es más que recomendable, porque añade una capa de seguridad que impide accesos no autorizados incluso si alguien descubre una contraseña.

Pero IAM no se basa solo en usuarios. También puedes trabajar con **roles**, que son como identidades temporales o delegadas. Un rol no está vinculado a una persona específica, sino a una función. Por ejemplo, puedes tener un rol llamado "BackupManager" que tiene permiso para acceder a los buckets de S3 y lanzar copias de seguridad, y ese rol puede ser asumido por una instancia EC2, una Lambda o incluso por otro usuario, según lo necesites. Este sistema es muy flexible y permite automatizar tareas o limitar accesos de forma muy precisa, algo especialmente útil en entornos grandes o cuando trabajas con arquitecturas de microservicios.

Y aquí llegamos a las **políticas**, que son el corazón del sistema de permisos en IAM. Una política es un documento en formato JSON (aunque AWS te lo muestra con asistentes para facilitar su uso) que define lo que está permitido o denegado. Las políticas se pueden asociar a usuarios, grupos o roles, y permiten controlar aspectos tan concretos como "este usuario puede leer archivos de este bucket S3, pero no puede escribir", o "esta Lambda puede acceder a la base de datos RDS entre las 9:00 y las 18:00". Es como escribir un contrato digital donde se detalla cada derecho y restricción. Existen políticas gestionadas por AWS (ya configuradas y probadas) y políticas personalizadas, que puedes crear desde cero según las necesidades específicas de tu proyecto.

Ahora bien, con tantas opciones, es fácil cometer errores. Por eso, conviene seguir **algunas buenas prácticas básicas** desde el principio. La primera es aplicar el principio de **mínimos privilegios**: cada usuario o servicio debe tener solo los permisos estrictamente necesarios para hacer su trabajo. Nada más. Si alguien necesita consultar un bucket, no le des acceso total a todo S3. Esta práctica reduce el riesgo de errores y limita los daños en caso de que una credencial se vea comprometida.

Ejemplo

Backups automáticos seguros en AWS

El objetivo es realizar copias de seguridad diarias desde una instancia EC2 a un bucket S3, usando usuarios, roles y políticas IAM con seguridad reforzada.

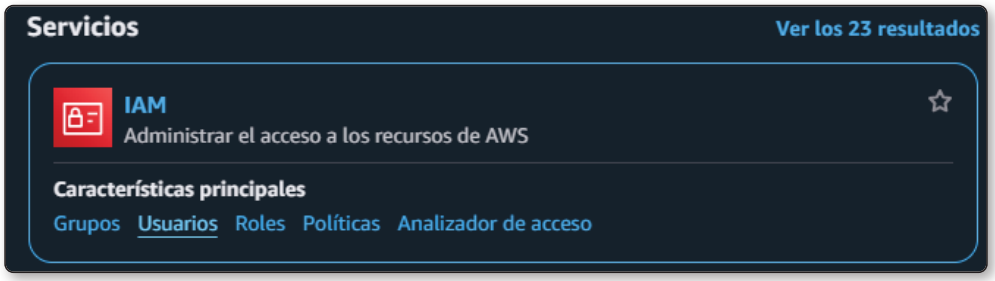
Paso 1: crear un usuario para administración

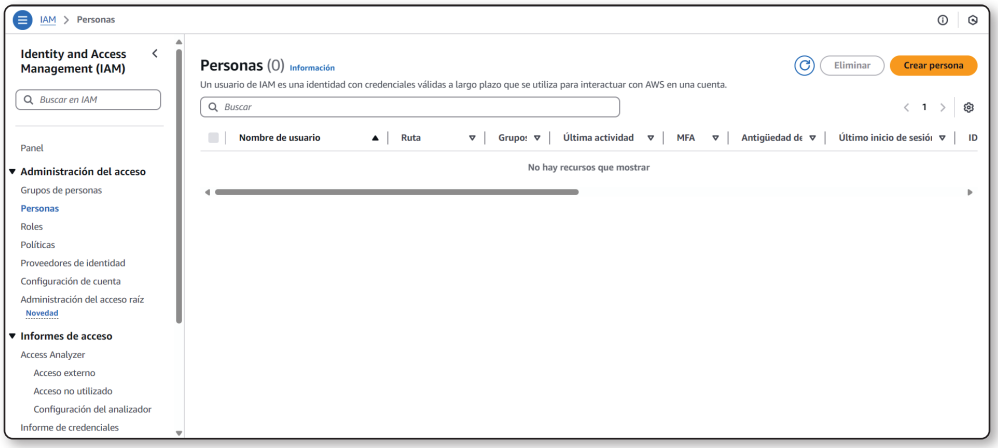
Nombre del usuario: Admin_Backups

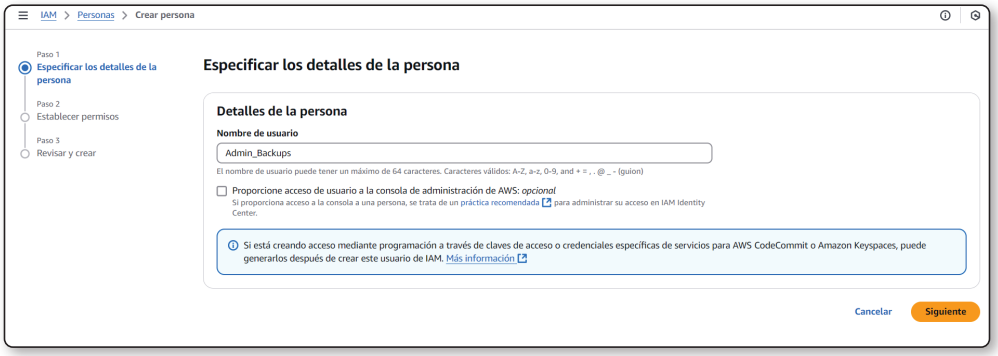
Uso: persona que configura y revisa las copias, pero no puede borrar nada.

Cómo hacerlo:

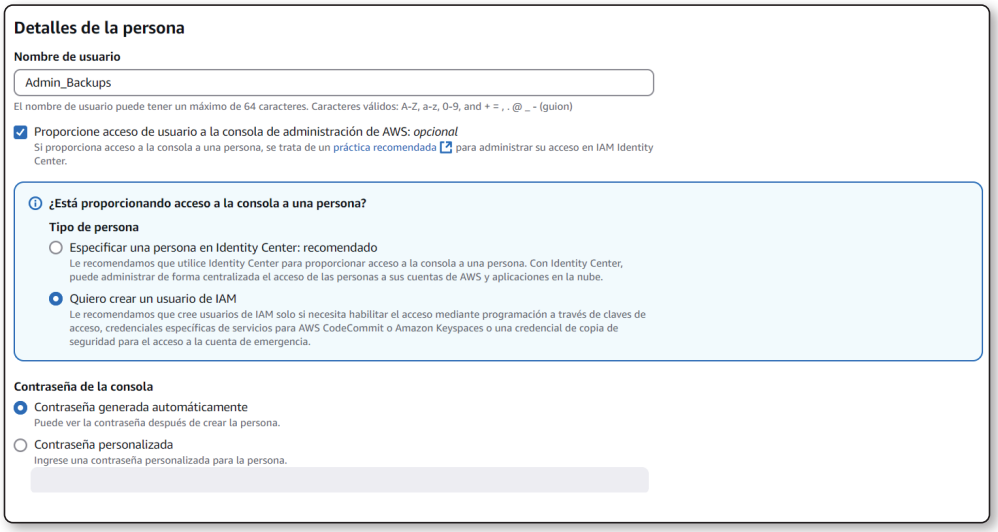
- Ve a IAM → "Usuarios" → "Crear persona".







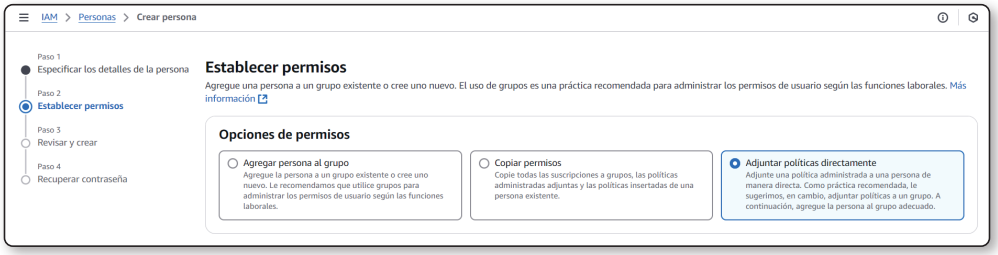
- Marca acceso a la consola de AWS.



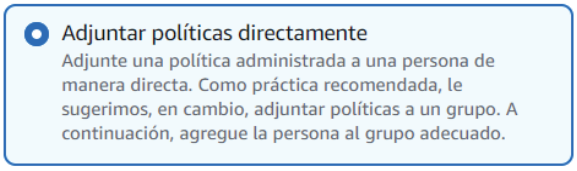
- Asigna una contraseña fuerte (En este caso se ha elegido la opción de generar la contraseña automáticamente).
- Activa MFA con una app como Google Authenticator.

Asigna políticas:

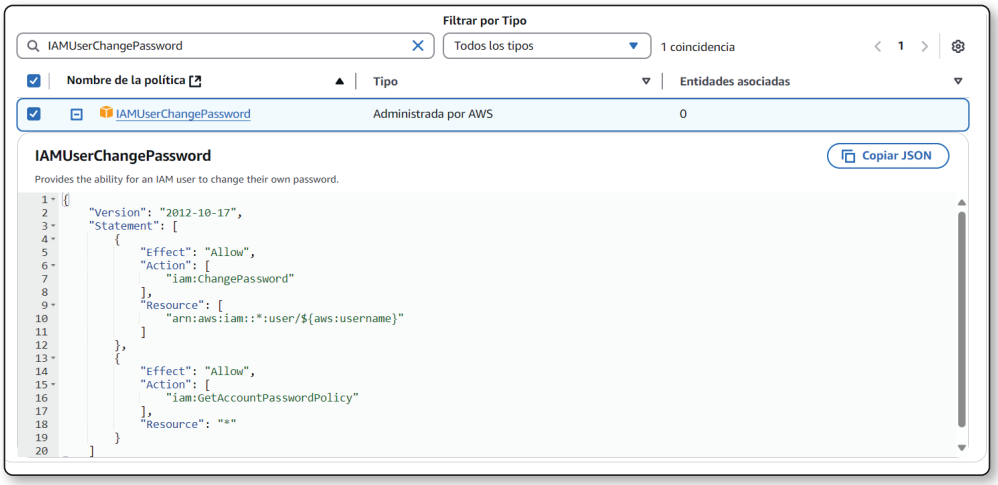
- El siguiente paso es establecer permisos:



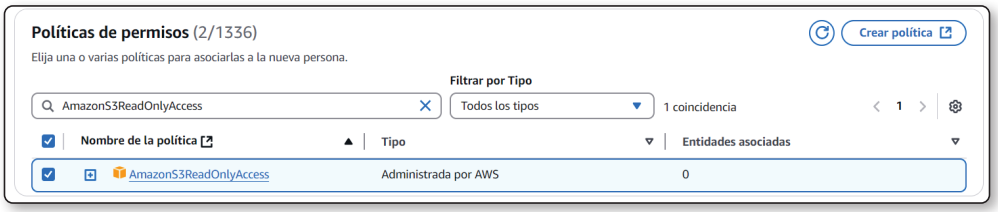
- Se va a seleccionar “Adjuntar políticas directamente”:



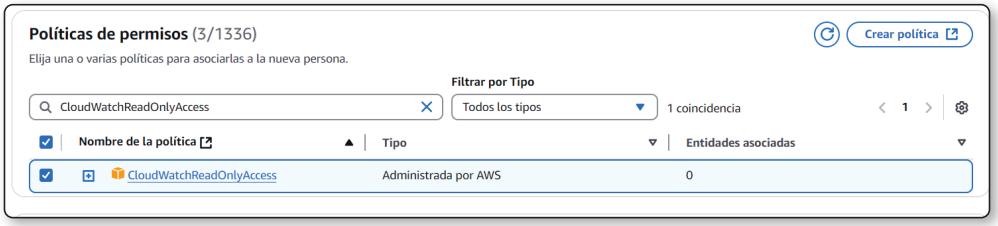
- En este caso vamos a marcar las siguientes:
  - IAMUserChangePassword:



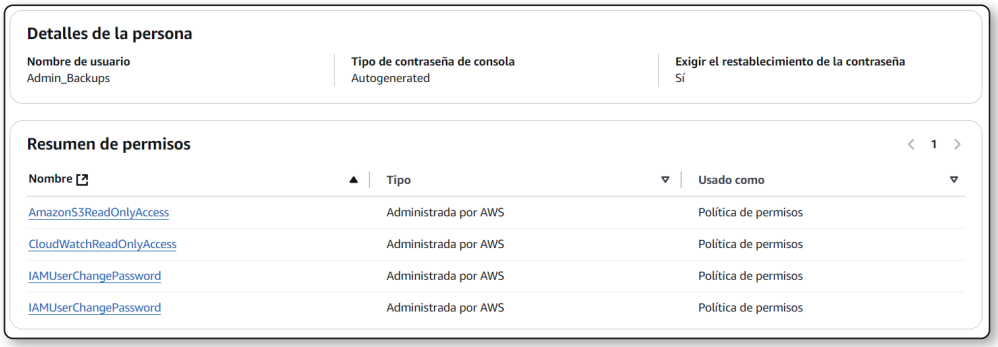
  - Acceso de solo lectura a S3 (AmazonS3ReadOnlyAccess):

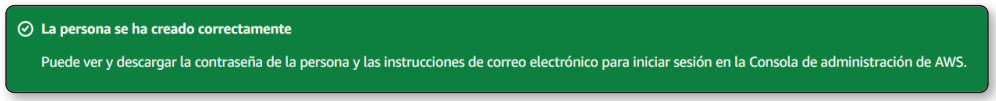


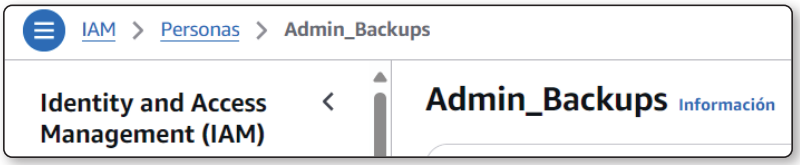
- Acceso a logs con CloudWatchReadOnlyAccess:



Este usuario puede revisar y configurar, pero no borrar.



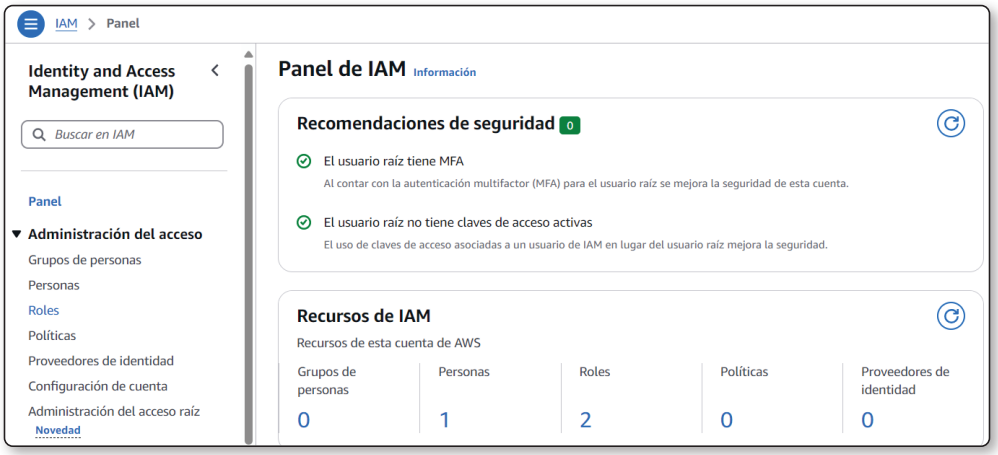


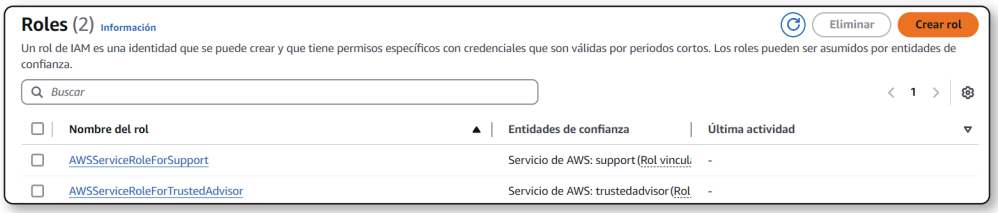


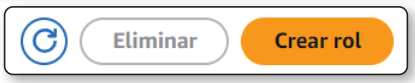
Paso 2: crear un rol para la EC2

Nombre del rol: EC2_BackupManager
Uso: permitir a la EC2 subir backups al bucket S3 sin más permisos.
Cómo hacerlo:

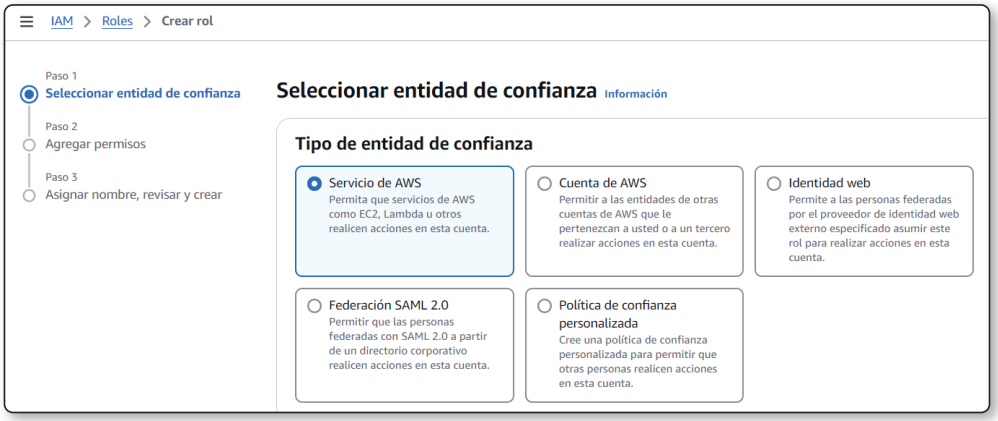
- Ve a IAM → "Roles" → "Crear rol".

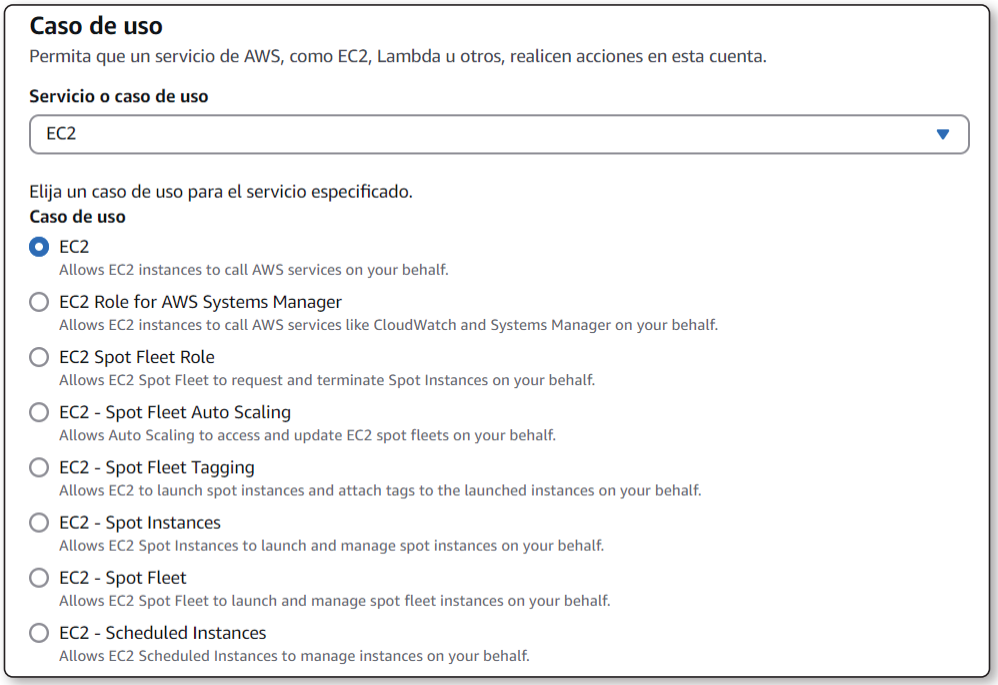




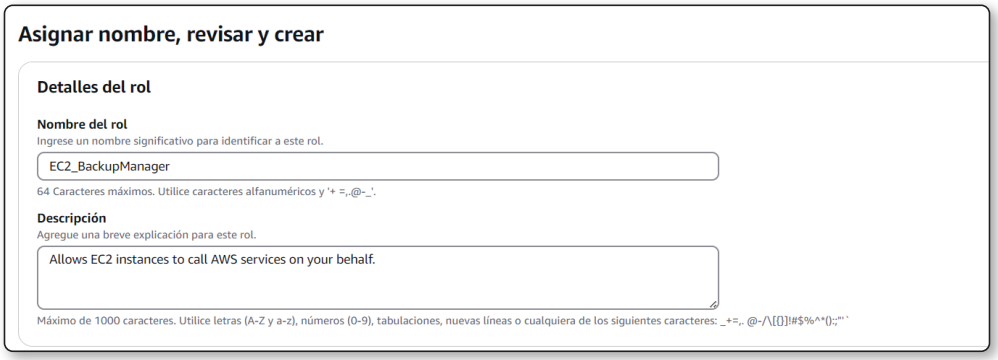


- Selecciona "EC2" como tipo de entidad.

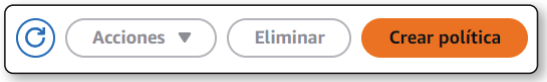




- Haz clic en el botón azul que dice “Siguiente” (esquina inferior derecha), sin seleccionar ninguna política todavía. Después pon el nombre al rol y selecciona “Crear rol”:



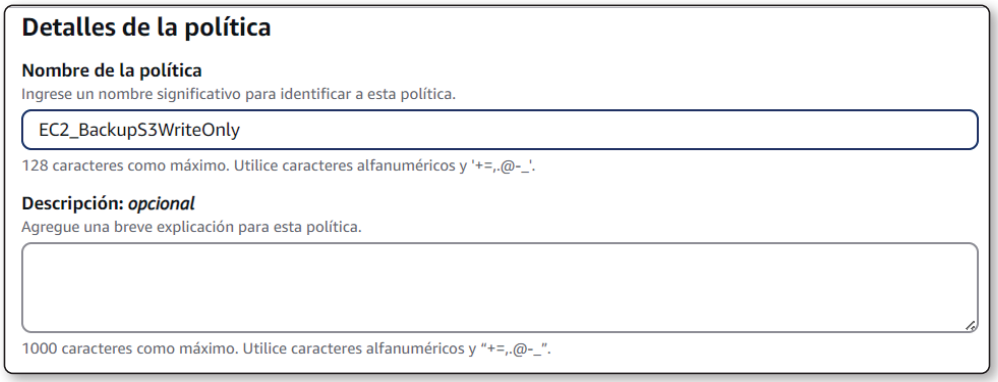
- Añade una política personalizada para subir backups a S3:
  Para ello:
  - Ve al servicio IAM → Políticas (en el menú lateral izquierdo).
  - Haz clic en "Crear política".



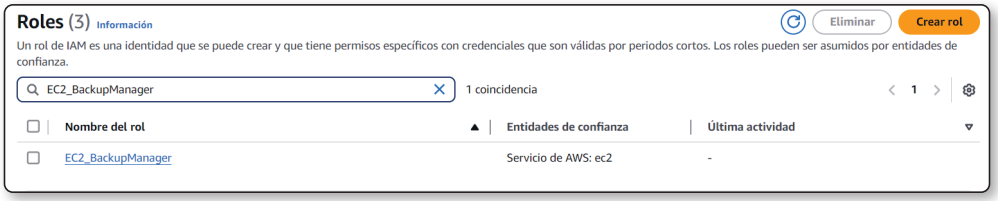
  - En la pestaña "Editor JSON", copia y pega este código:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["s3:PutObject"],
      "Resource": "arn:aws:s3:::backups-guttman/*"
    }
  ]
}
```

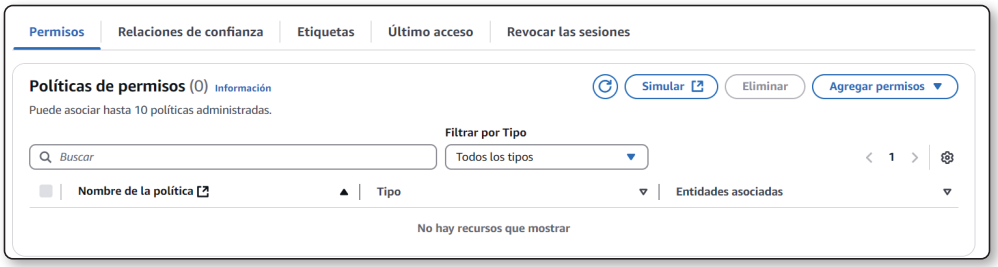
  - Haz clic en "Siguiente", ponle un nombre (por ejemplo: EC2_BackupS3WriteOnly) y finaliza la creación.

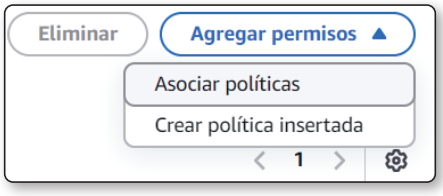


- Añadir esta política al rol que ya creaste:
  - Ve a IAM → Roles.
  - Busca el rol EC2_BackupManager.

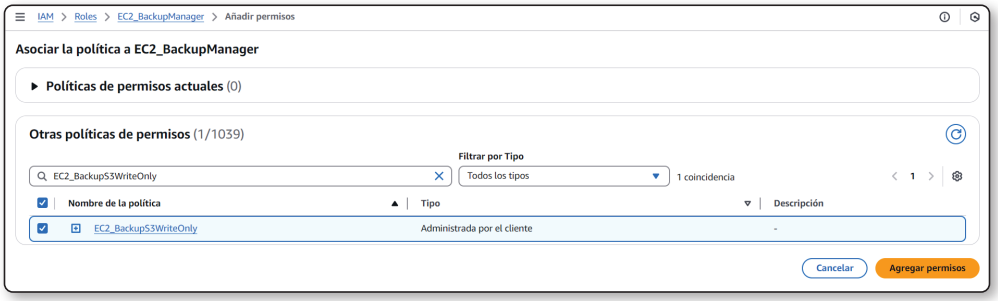


  - Marca la casilla y haz clic en el rol para acceder a las políticas de permisos y selecciona "Agregar permisos" > "Asociar políticas".





  - Busca la política que acabas de crear (EC2_BackupS3WriteOnly), márcala y haz clic en "Agregar permisos".



Paso 3: aplicar el principio de mínimos privilegios

Este paso asegura que cada entidad solo tenga los permisos estrictamente necesarios.

Usuario Admin_Backups

- Solo tiene permisos de lectura sobre S3 (AmazonS3ReadOnlyAccess): no puede borrar, modificar ni subir archivos.
- Puede ver logs (CloudWatchReadOnlyAccess): necesario para auditar, pero sin capacidad de cambio.
- Puede cambiar su contraseña (IAMUserChangePassword): mínimo necesario para autogestión.
- Tiene MFA activo: añade una capa extra de seguridad ante accesos indebidos.
- No tiene permisos administrativos generales ni acceso completo a IAM ni S3.

Conclusión: perfecto para un perfil humano de supervisión, sin riesgo operativo.

Rol EC2_BackupManager

- Solo tiene la política personalizada de escritura (s3:PutObject) sobre el bucket backups-guttman/*.
- No puede leer, listar, borrar ni acceder a ningún otro bucket ni recurso.
- El rol se asignó a una instancia EC2 concreta, lo que limita su alcance aún más.

Conclusión: el rol está diseñado para hacer una única cosa: escribir copias de seguridad. Esto es un ejemplo excelente de privilegios mínimos.

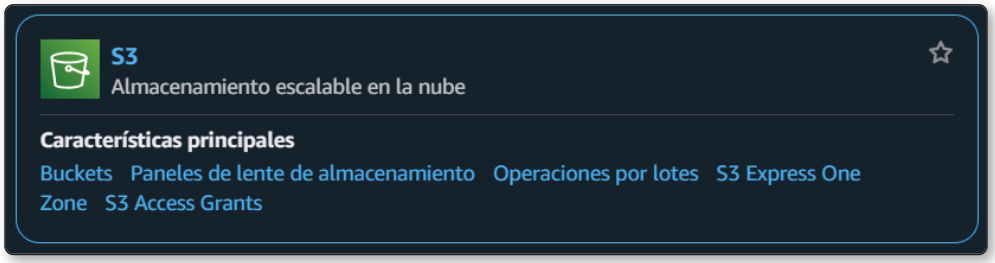
También es buena idea **organizar a los usuarios por grupos**, sobre todo si gestionas varios perfiles. Por ejemplo, puedes tener un grupo "Desarrolladores" con acceso a EC2 y S3, y otro grupo "Administradores" con acceso más amplio. Así, si se incorpora alguien nuevo al equipo, basta con añadirlo al grupo adecuado, sin tener que configurar sus permisos desde cero.

Otro consejo importante: **activa la autenticación multifactor** (MFA) para todos los accesos sensibles, especialmente para el usuario root y para quienes tienen permisos de administración. Además, **revisa periódicamente las políticas activas**, y elimina las que ya no se usen o los accesos temporales que hayan caducado. AWS ofrece herramientas como **IAM Access Analyzer** que ayudan a detectar políticas demasiado amplias o accesos innecesarios.´

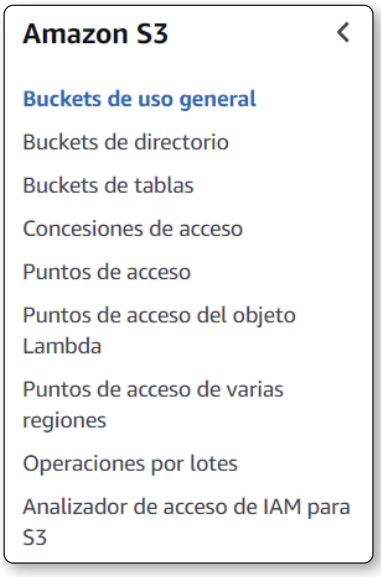
IAM también se integra con otros servicios de control de identidad, como **AWS Organizations**, que permite gestionar varias cuentas desde una misma consola, o con servicios de directorio como **AWS Directory Service**, ideal para entornos empresariales donde se quiera usar Active Directory para autenticar usuarios. Incluso puedes permitir acceso federado desde plataformas externas, como Google Workspace o Azure AD, mediante **federación SAML**, algo que facilita mucho la integración en organizaciones grandes.

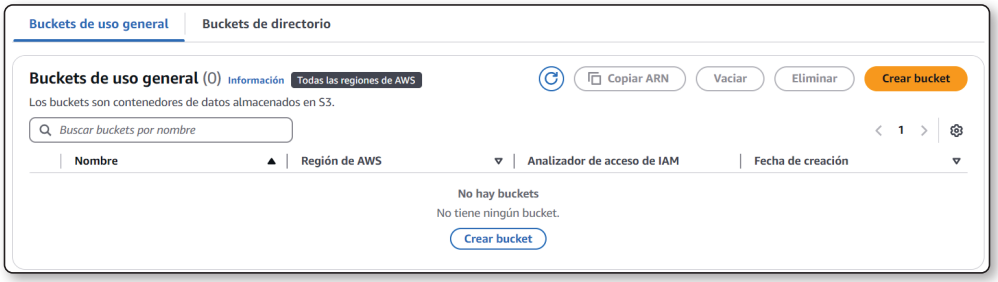
3.3 ALMACENAMIENTO CON AMAZON S3

Cuando se empieza a trabajar con AWS, tarde o temprano hay que cruzarse con Amazon S3. Su nombre completo es *Amazon Simple Storage Service*, y aunque suena largo, la idea es muy sencilla: guardar archivos de forma segura, escalable y accesible desde cualquier parte del mundo. Dicho de otra forma, S3 es como un disco duro en la nube, pero con esteroides. No tiene límite práctico de capacidad, puedes almacenar desde una imagen hasta terabytes de datos, y todo queda perfectamente organizado si se siguen unas reglas claras. Es uno de los servicios más usados en todo tipo de proyectos: desde páginas web personales hasta plataformas de streaming, pasando por aplicaciones móviles, copias de seguridad o sistemas de inteligencia artificial que necesitan grandes volúmenes de información.



La unidad básica de almacenamiento en S3 es el **bucket**, que podríamos traducir como "cubo". Un bucket es, en realidad, un contenedor donde se guardan los objetos. Y cuando hablamos de objetos nos referimos a archivos de todo tipo: imágenes, vídeos, documentos, bases de datos, lo que quieras. Cada bucket tiene un nombre único a nivel global, lo que significa que no puedes crear un bucket llamado *misfotos* si alguien en otro lugar del mundo ya ha registrado ese nombre. Por eso, lo habitual es usar nombres personalizados o incluir un sufijo con el nombre de tu proyecto, como misfotos-trixal-2025.





Dentro de cada bucket puedes organizar los archivos con una estructura similar a la de carpetas, aunque técnicamente lo que hace S3 es manejar rutas y prefijos, no carpetas reales como las que ves en tu ordenador. Pero visualmente y en la práctica, puedes tener algo así como /documentos/facturas/2024/factura123.pdf, y navegar por esas "carpetas" desde la consola. Esta organización es muy flexible y facilita mucho el trabajo cuando se manejan miles de archivos.

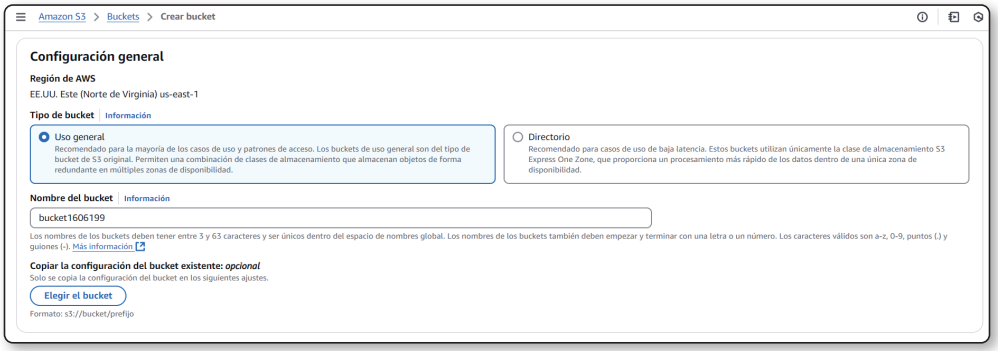
Recurso

Para crear un bucket

Haz clic en "Crear bucket"

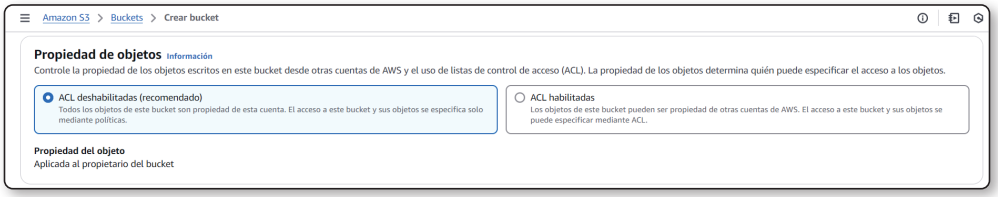
- Esto abrirá un formulario donde puedes configurar tu bucket.

Configura el nombre y región del bucket:



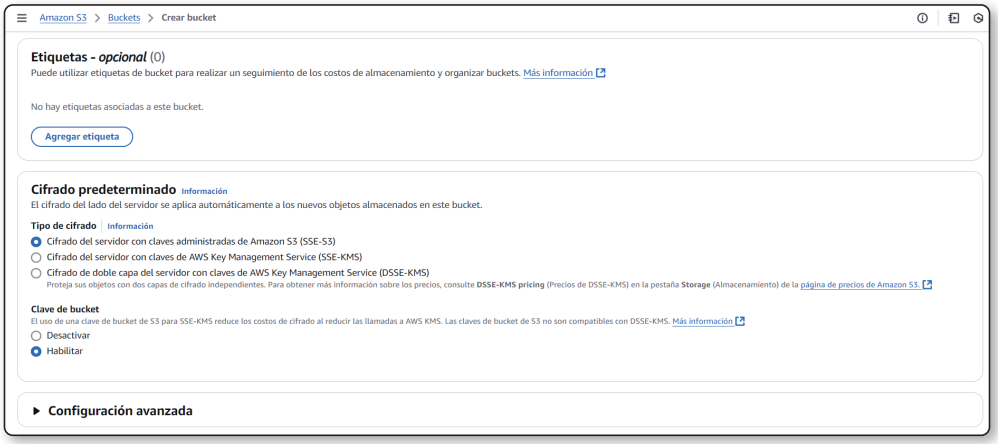
- Nombre del bucket: debe ser único a nivel global.
- Región: elige la región donde quieres almacenar tus datos (por ejemplo, eu-west-1 para Irlanda).

Configura opciones de seguridad:



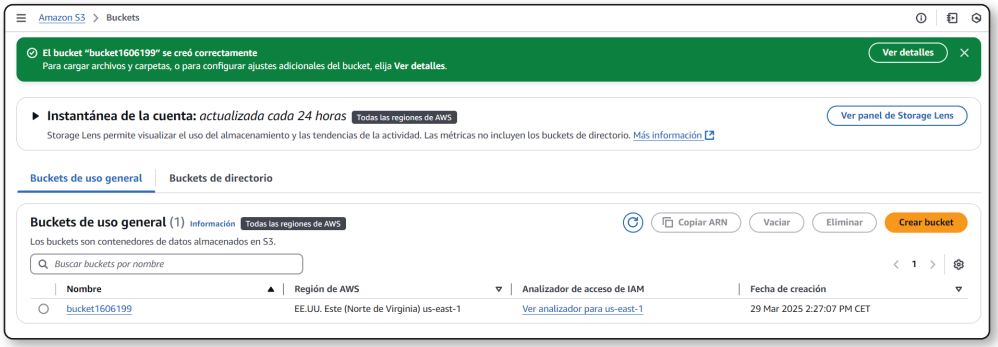
- Puedes desactivar el acceso público si quieres que solo ciertos usuarios tengan acceso.
- También puedes habilitar cifrado (por ejemplo, SSE-S3) si necesitas proteger la información.

Opcional: configura versiones, etiquetas y logs:



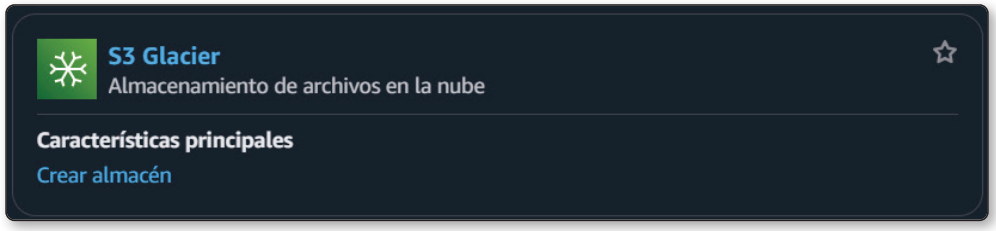
- Puedes activar el versionado (para conservar versiones antiguas), añadir etiquetas para organizar tus recursos o activar registros de acceso.

Revisa las opciones y haz clic en "Crear bucket":



Ahora bien, una de las cosas más potentes de S3 es que no te obliga a usar un solo tipo de almacenamiento para todos los archivos. Amazon ha diseñado **clases de almacenamiento**, que son básicamente modos distintos de guardar los datos según el uso que les vayas a dar. Por ejemplo, si tienes archivos que se consultan todos los días, lo ideal es usar la clase **Standard**, que ofrece baja latencia y alta disponibilidad. Es perfecta para aplicaciones activas, sitios web o sistemas en producción.

Pero si lo que tienes son documentos que necesitas guardar por obligación legal o simplemente por si acaso —como facturas antiguas, backups o registros de logs que nadie consulta a menudo—, puedes optar por otras clases como **S3 Glacier** o **S3 Glacier Deep Archive**, que son mucho más baratas porque están pensadas para datos que se acceden una o dos veces al año. La pega es que tardan más tiempo en estar disponibles cuando las necesitas, a veces horas.



Para archivos que se consultan ocasionalmente pero que aún deben estar accesibles rápidamente, existe la clase **S3 Standard-IA (Infrequent Access)**. Y si quieres que AWS decida automáticamente qué clase es mejor en función del uso real, puedes activar el **almacenamiento inteligente (Intelligent-Tiering)**, que va moviendo los objetos entre clases según su nivel de acceso, ahorrando dinero sin que tengas que preocuparte por ello.

Una de las ventajas más interesantes de S3 es el **control de acceso**, porque no todo lo que subes tiene que estar disponible para todo el mundo. De hecho, por defecto, todo lo que guardas en un bucket es privado. Solo tú (o quienes tú autorices) pueden ver y manipular esos archivos. El sistema de permisos de S3 es muy completo y se puede gestionar a varios niveles. Puedes dar permisos a nivel de bucket completo o a nivel de objeto específico. Por ejemplo, puedes permitir que una imagen concreta esté disponible públicamente en tu web, pero mantener todo lo demás bajo llave.

Los permisos se gestionan mediante **políticas**, que pueden ser políticas de bucket (afectan a todos los archivos dentro del cubo) o políticas de acceso a nivel de usuario usando IAM. También puedes generar URLs temporales que dan acceso a un archivo durante un tiempo limitado, lo cual es muy útil si quieres compartir algo con un cliente o con otro equipo sin dejarlo abierto indefinidamente. Y si trabajas con aplicaciones o usuarios externos, puedes usar el sistema **ACL (Access Control List)** para definir permisos más granulares.

Nota

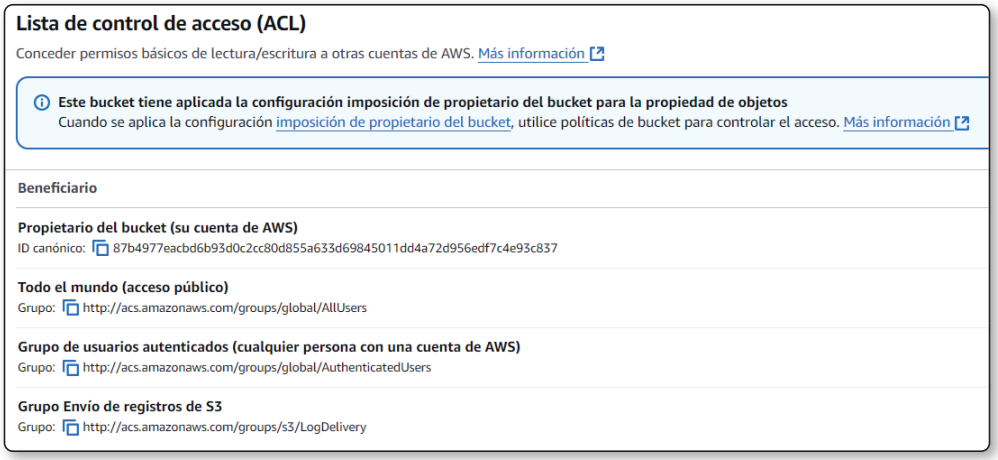
Recientemente, AWS recomienda gestionar el acceso mediante políticas de bucket (Bucket Policies) o políticas basadas en identidad (IAM Policies), ya que ofrecen mayor flexibilidad y seguridad.

¿Dónde encontrar la ACL en AWS?

Si necesitas específicamente acceder a la ACL de un bucket en S3, sigue estos pasos:

1. Accede a la Consola de AWS (https://console.aws.amazon.com).
2. Entra en el servicio Amazon S3.
3. Selecciona tu bucket haciendo clic en su nombre.
4. Ve a la pestaña “Permisos” en la parte superior del menú del bucket.

5. Desplázate hacia abajo hasta que veas la sección "Listas de control de acceso (ACL)":



Ahí verás las opciones clásicas para definir quién puede leer o escribir en el bucket.

Desde aquí podrás ver, editar o modificar la ACL

AWS ahora considera que las ACL son un método obsoleto para controlar el acceso.

Recomiendan usar Bucket Policies o políticas basadas en IAM porque son más seguras y manejables.

Por ejemplo, si tienes este escenario:

- Dar acceso público de solo lectura → Usa Bucket Policy.
- Dar acceso a usuarios específicos o roles IAM → Usa IAM Policies.

Las ACL quedan principalmente para casos concretos o de compatibilidad con aplicaciones antiguas.

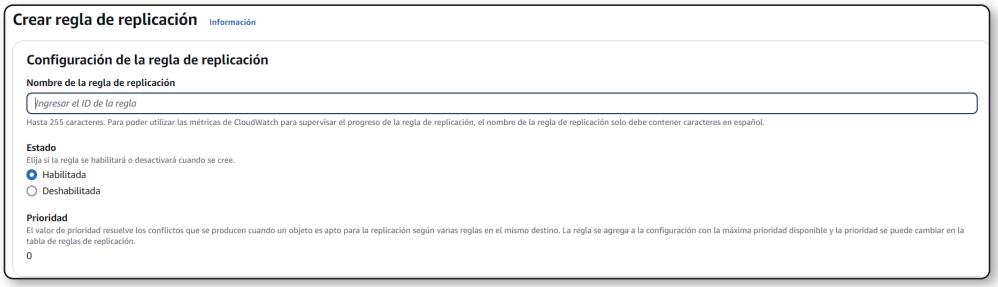
S3 también te permite configurar reglas para **versionado de archivos**, lo cual es muy útil cuando necesitas mantener un historial de cambios. Activando el versionado, cada vez que subes un archivo con el mismo nombre, S3 guarda una versión nueva sin borrar la anterior. Así puedes volver atrás si algo sale mal o si necesitas recuperar un archivo borrado accidentalmente. Otra opción interesante es la **replicación entre regiones**, que te permite mantener copias automáticas de tus datos en otra región geográfica, lo que mejora la resiliencia ante desastres o fallos regionales.

Recurso

Para acceder a la replicación entre regiones debes hacer clic en la pestaña "Administración".

Una vez allí, encontrarás fácilmente la opción **"Crear regla de replicación"**, donde puedes configurar la replicación entre regiones.

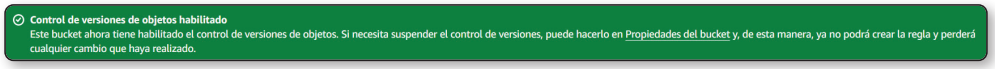
Desde ahí podrás añadir reglas y configurar la replicación entre regiones.



1. Nombre de la regla de replicación
   - Aquí simplemente escribe un nombre identificativo, por ejemplo:
   - *replicacion-region-backup*
2. Estado
   - Déjalo en "Habilitada" si quieres que la replicación empiece a funcionar inmediatamente tras crearla.
3. Prioridad
   - Si tienes varias reglas de replicación que podrían afectar a los mismos objetos, esta prioridad determina cuál regla prevalece.
   - Por defecto puedes dejarlo en 0 si esta es la única regla o la más importante.

Completa los siguientes detalles según tus necesidades y luego continúa hasta completar la creación.

*Es importante mencionar que el bucket debe tener habilitado el control de versiones de objeto para crear esta regla:



Desde el punto de vista técnico, S3 destaca por su **durabilidad y disponibilidad**. Amazon garantiza una durabilidad del 99.999999999% (11 nueves), lo que significa que es extremadamente poco probable que pierdas un archivo. Esto lo consigue replicando automáticamente cada objeto en múltiples dispositivos dentro de una región. Además, puedes configurar notificaciones para saber cuándo alguien sube, borra o modifica un archivo, lo cual es muy útil para automatizar flujos de trabajo.

Además de todas sus funcionalidades básicas, Amazon S3 ofrece un abanico de herramientas complementarias que permiten sacarle el máximo partido a su arquitectura, sobre todo cuando se empieza a trabajar con grandes volúmenes de datos o se quiere automatizar la gestión del almacenamiento. Por ejemplo, una opción muy interesante es la **gestión del ciclo de vida**, que permite definir reglas automáticas para mover objetos entre distintas clases de almacenamiento o eliminarlos tras un

periodo determinado. Esta característica es ideal en proyectos donde los datos tienen una vigencia clara. Por ejemplo, si tienes registros que solo necesitas guardar durante tres meses, puedes programar su eliminación automática y olvidarte del asunto. Del mismo modo, puedes establecer que pasen de S3 Standard a Standard-IA tras 30 días de inactividad, y después a Glacier si no se accede a ellos en un año. Esto optimiza los costes sin comprometer la organización.

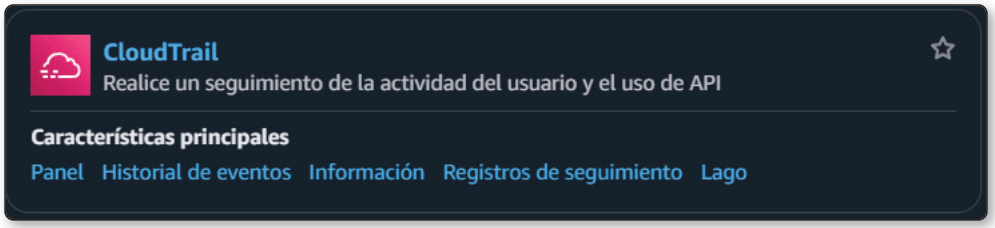
Otra característica poco conocida, pero muy útil, es **S3 Object Lock**, una función pensada para cumplir con normativas estrictas de retención de datos, como las que afectan al sector financiero o sanitario. Con Object Lock puedes asegurarte de que un archivo no se borrará ni modificado durante un tiempo específico. Esto añade una capa de protección legal, técnica y operativa que muchas empresas necesitan para estar tranquilas en auditorías o inspecciones. No es algo que se use todos los días, pero cuando lo necesitas, se vuelve esencial.

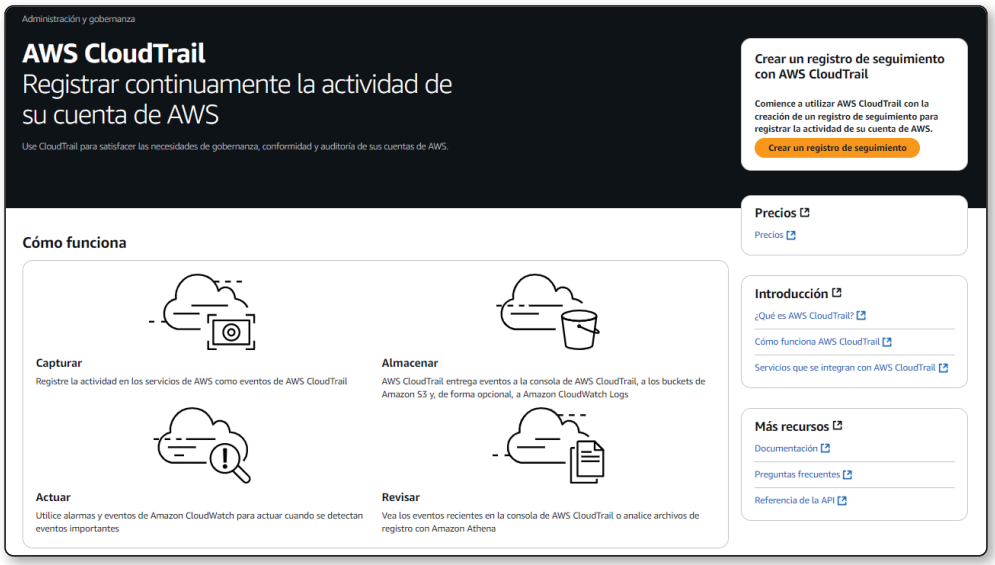
S3 también puede integrarse fácilmente con otros servicios de AWS, lo que permite crear flujos de trabajo muy potentes sin tener que salir del entorno. Por ejemplo, puedes conectar un bucket con **Lambda**, el servicio que ejecuta código sin necesidad de servidores, y hacer que cada vez que se suba un archivo se ejecute automáticamente una función: convertir un vídeo a otro formato, procesar una imagen, enviar una notificación... Las posibilidades son enormes y abren la puerta a una automatización que antes solo era posible con servidores dedicados o código complejo.

Además, Amazon S3 puede funcionar como origen estático para una página web. Esto significa que puedes alojar sitios web sencillos directamente desde un bucket, sin necesidad de configurar servidores ni sistemas complejos. Solo tienes que subir tus archivos HTML, CSS, imágenes y demás, activar la opción de "hosting estático" y ya tienes una web disponible para el mundo. Esto lo usan muchos desarrolladores que quieren mostrar portfolios, sitios personales o pequeñas landing pages de forma rápida y sin gastos.

También se pueden aplicar reglas de **cifrado automático**, tanto en reposo como en tránsito. Esto quiere decir que los datos que almacenes estarán protegidos en todo momento, incluso si alguien logra acceder de forma no autorizada a la infraestructura física. AWS ofrece opciones de cifrado gestionado por el propio servicio (SSE-S3), con claves gestionadas por el cliente (SSE-C), o mediante el servicio KMS (AWS Key Management Service), que permite llevar un control más detallado de las claves. Estas opciones cumplen con los estándares internacionales y son compatibles con las exigencias legales en contextos europeos como el RGPD.

Por último, no hay que olvidar que todo lo que pasa en S3 puede ser monitorizado y auditado. Usando servicios como **AWS CloudTrail**, puedes ver quién accedió a qué archivo, cuándo lo hizo y desde dónde. Esto es especialmente útil para proyectos colaborativos o entornos empresariales donde varias personas o sistemas interactúan con los mismos datos. La trazabilidad de las acciones aporta una capa de seguridad y responsabilidad que, bien configurada, puede ser decisiva para detectar comportamientos anómalos o garantizar el cumplimiento de políticas internas.





En conjunto, Amazon S3 no es simplemente un lugar para subir archivos. Es un sistema diseñado para acompañar el crecimiento de los proyectos, adaptarse a distintos contextos legales y técnicos, facilitar la automatización y, sobre todo, proporcionar un entorno de almacenamiento fiable y escalable que se ajusta tanto a pequeñas pruebas personales como a despliegues empresariales de gran envergadura.

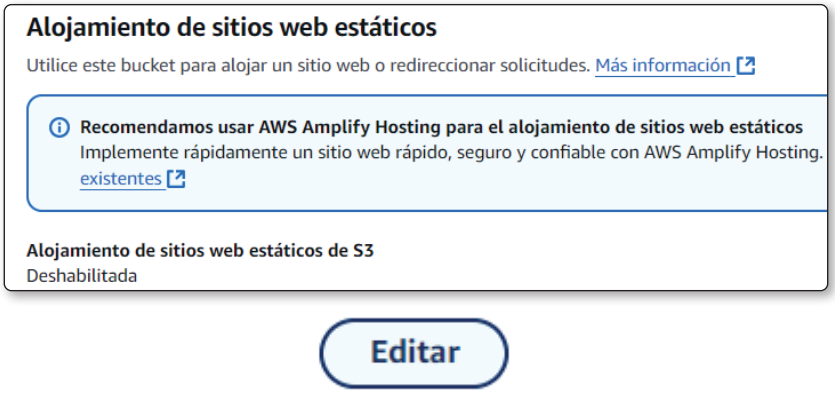
3.4 WEBSITE ESTÁTICO

Montar una página web estática puede parecer algo del pasado si se compara con las aplicaciones dinámicas llenas de interactividad que dominan hoy en día. Sin embargo, las webs estáticas siguen siendo una opción muy interesante cuando lo que se necesita es algo sencillo, rápido, seguro y fácil de mantener. Sitios personales, portfolios de diseñadores, páginas informativas de proyectos, landing pages o presentaciones de productos son algunos ejemplos donde una web estática cumple perfectamente su función. Y lo mejor es que gracias a servicios como **Amazon S3** y **CloudFront**, hoy es posible alojar este tipo de sitios con altísima disponibilidad, excelente rendimiento y sin complicaciones técnicas.

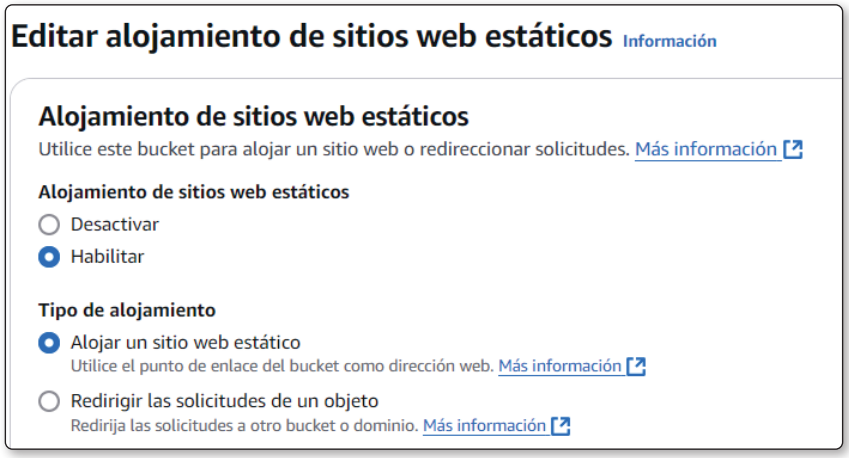
El proceso de alojar una web estática en **Amazon S3** es bastante intuitivo, incluso para quienes no tienen mucha experiencia técnica. Básicamente, se trata de subir los archivos que forman tu sitio web —HTML, CSS, JavaScript, imágenes, fuentes, etc.— a un **bucket** de S3. Este bucket funcionará como el contenedor de

tu página. Una vez que están dentro, hay que activar una opción específica llamada *static website hosting* (Alojamiento de sitios web estáticos), que convierte ese bucket en un punto de acceso público donde cualquiera podrá ver tu web. Luego defines el archivo de inicio (generalmente index.html) y, si lo necesitas, también una página de error personalizada (error.html). A partir de ahí, tu sitio está publicado.

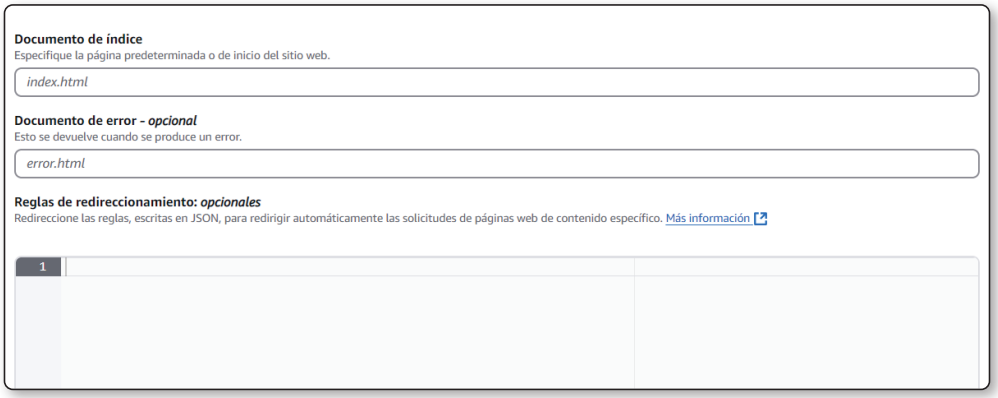
Para acceder al alojamiento de sitios web estáticos hay que ir a "Propiedades"



Editar sitio web estático



Habilitar alojamiento de sitios web estáticos



Completar según requerimientos propios

Pero hay que tener en cuenta que, por defecto, los buckets de S3 están configurados para ser privados. Esto es lógico por motivos de seguridad. Para que tu sitio esté disponible para todo el mundo, debes modificar las **políticas de acceso** del bucket y permitir la lectura pública de los objetos. Esta parte hay que hacerla con cuidado, solo habilitando lo estrictamente necesario, porque abrir demasiado puede exponer tus datos si no gestionas bien los permisos.

Una vez publicado tu sitio, si el público objetivo está muy repartido geográficamente o si esperas que muchas personas accedan al mismo tiempo, es recomendable mejorar su rendimiento mediante **Amazon CloudFront**. Este servicio es una **red de distribución de contenido (CDN)** que se encarga de servir tu página desde el lugar más cercano al visitante. ¿Qué significa esto en la práctica? Que si alguien entra a tu web desde Sevilla y otro desde Lima, cada uno accederá a una copia almacenada temporalmente en un nodo cercano, lo que reduce el tiempo de carga y mejora la experiencia de navegación.

CloudFront funciona almacenando en **edge locations** (puntos de presencia distribuidos por todo el mundo) copias del contenido de tu sitio. Estos puntos se actualizan periódicamente para mantener sincronizados los archivos con la versión original en S3. Así, si tienes una imagen en tu web que pesa 1 MB, no se descargará siempre desde el bucket original, sino desde la caché más cercana. Esto no solo acelera la carga, también reduce el tráfico directo a S3, lo que puede ayudarte a mantener los costes bajo control, sobre todo si estás en la capa gratuita.

Configurar CloudFront con un bucket de S3 requiere algunos pasos más técnicos, como crear una **distribución** y enlazarla correctamente al origen del contenido. También puedes configurar un **nombre de dominio personalizado** (por ejemplo, www.misitio.com) usando **Route 53** o cualquier otro proveedor de DNS. Para que el sitio funcione bajo **HTTPS** —algo imprescindible hoy en día por seguridad y por posicionamiento en buscadores—, puedes utilizar certificados SSL gratuitos mediante **AWS Certificate Manager**. Todo esto puede parecer mucho al principio, pero una vez hecho, tu sitio está preparado para funcionar como una web profesional con velocidad y protección de primer nivel.

Otra ventaja interesante es que al estar alojado en S3, tu sitio web es muy **resistente a fallos**. Al no depender de un servidor tradicional que pueda caerse, sino de una infraestructura distribuida y altamente redundante, es muy poco probable que sufra interrupciones. De hecho, muchas empresas eligen esta solución para mostrar mensajes de mantenimiento cuando sus sistemas principales están caídos, porque saben que la web estática se mantendrá online pase lo que pase.

ⓘ Nota

Usar **Amazon S3 y CloudFront para alojar un sitio estático** permite montar una web profesional sin tener que gestionar servidores, sin preocuparse por las actualizaciones del sistema operativo y con la posibilidad de escalar automáticamente si el tráfico se dispara. Es una opción perfecta para quienes quieren enfocarse en el contenido sin complicarse con la infraestructura, y para quienes valoran la rapidez, la seguridad y la fiabilidad que ofrece el ecosistema de AWS. Una combinación sencilla en apariencia, pero muy potente en resultados.

Otro aspecto especialmente interesante del alojamiento de sitios estáticos en Amazon S3 es su integración con flujos de trabajo automatizados. En lugar de subir manualmente los archivos cada vez que haces un cambio, puedes conectarlo con herramientas de desarrollo modernas que facilitan el **despliegue continuo**. Por ejemplo, si usas GitHub, puedes configurar un pipeline que detecte cuando actualizas el repositorio y lance una nueva versión de tu sitio directamente al bucket de S3. Esto no solo ahorra tiempo, sino que asegura que todos los cambios estén reflejados al instante y sin errores humanos durante la carga de archivos.

En el contexto educativo o de pequeños proyectos empresariales, esta forma de trabajar también representa una solución accesible y con un mantenimiento muy bajo. No necesitas licencias, ni comprar máquinas físicas, ni preocuparte por parches de seguridad. Además, puedes controlar el **versionado de tu contenido**, ya que S3

permite conservar versiones anteriores de los archivos, por si necesitas volver atrás en caso de que algo no funcione como esperabas.

Respecto a los costes, esta arquitectura es especialmente eficiente. La capa gratuita de AWS te permite almacenar hasta **5 GB de archivos en S3** y servir una gran cantidad de peticiones mensuales sin coste. Combinado con CloudFront, puedes tener un sitio web disponible en todo el mundo, con tiempos de carga muy bajos y sin pagar un euro durante bastante tiempo si el tráfico es moderado. Y cuando llega el momento de pasar a una escala mayor, solo se paga por lo que se usa: almacenamiento adicional, número de peticiones o ancho de banda consumido.

Una cuestión que se valora mucho, sobre todo en proyectos institucionales o públicos en España, es el cumplimiento de normativas y la gestión del contenido estático bajo control europeo. Gracias a la posibilidad de **elegir regiones específicas** para almacenar los datos (como eu-west-3 en París o eu-south-1 en Milán), se puede asegurar que los ficheros del sitio estén dentro del Espacio Económico Europeo. Esto facilita el cumplimiento del RGPD, especialmente cuando se alojan documentos, formularios u otros recursos informativos que puedan contener datos personales o sensibles.

Por otro lado, los mecanismos de **control de acceso** en S3 y CloudFront permiten limitar o abrir el acceso de forma muy detallada. Puedes hacer que toda tu web sea pública, pero también puedes proteger determinadas carpetas o archivos con claves temporales, tokens firmados o cabeceras específicas. Esto resulta muy útil cuando se quieren alojar recursos de uso interno o que solo estén disponibles para un grupo de personas, como contenidos premium, manuales corporativos o secciones para usuarios registrados. La flexibilidad en este sentido es muy alta, y puedes construir sistemas de acceso personalizados sin necesidad de tener un backend complejo.

En términos de rendimiento, usar S3 y CloudFront también tiene un impacto positivo en el **SEO** y la experiencia de usuario. Las páginas estáticas, al no depender de bases de datos ni de generación dinámica, se cargan en tiempos mínimos, lo que mejora la percepción del visitante y reduce el porcentaje de abandono. Además, es más fácil optimizar aspectos como la compresión de archivos, el uso de caché o la entrega de recursos comprimidos (gzip o Brotli), lo cual mejora aún más el rendimiento general del sitio. Todo esto es clave en proyectos donde la velocidad de carga marca la diferencia, como portales de campañas digitales, portafolios artísticos o webs promocionales.

Finalmente, una ventaja poco mencionada es la capacidad de usar **funciones de redirección y error personalizadas** dentro del propio S3. Por ejemplo, puedes definir qué hacer si alguien intenta acceder a una página que no existe, redirigir tráfico hacia una URL externa o mostrar una página informativa con diseño propio. Esto da un toque más profesional al sitio, mejora la navegación y permite mantener una experiencia coherente sin depender de programación adicional. Y si se combina con las funcionalidades de **CloudFront Functions** o **Lambda@Edge**, incluso es posible ejecutar pequeñas funciones de lógica en el borde, como redirecciones condicionales, cambios en encabezados o personalización de contenido.

ⓘ Nota

El alojamiento de páginas web estáticas con Amazon S3 y CloudFront es una solución moderna, robusta y muy adaptable a diferentes necesidades. Desde proyectos personales hasta sitios institucionales, ofrece una alternativa sólida a los hosting tradicionales, con ventajas en coste, rendimiento, mantenimiento y cumplimiento normativo. Todo sin necesidad de gestionar servidores ni lidiar con complejas configuraciones. Una puerta de entrada muy potente para quienes quieran ofrecer contenidos digitales con calidad profesional desde el primer momento.