

03. S3 Replication

La réplication S3 est un mécanisme **asynchrone** et **automatique** permettant de copier des objets d'un bucket source vers un ou plusieurs buckets de destination.

1. Pourquoi utiliser la réplication ? (Business & Architecture Cases)

En architecture cloud, on ne duplique pas la donnée sans une excellente raison réglementaire ou technique. Voici la logique derrière chaque cas d'usage :

- **Plan de reprise d'activité (Disaster Recovery)** : Si une région AWS entière devenait indisponible (scénario catastrophe), l'application doit pouvoir basculer sur une copie des données opérationnelles située dans une autre région géographique.
 - **Conformité légale et souveraineté** : Certaines industries (banque, santé) obligent légalement à stocker des copies de sauvegarde à une distance minimale du centre de données principal, ou à l'inverse, à confiner strictement les données dans les frontières d'un même pays.
 - **Sécurité absolue (Isolation des comptes)** : Pour se prémunir d'un ransomware ou d'un employé malveillant qui obtiendrait les accès administrateur du compte principal, on réplique les données vers un bucket appartenant à un **autre compte AWS** totalement isolé, avec des clés d'accès différentes.
 - **Optimisation des performances (Latence)** : Si une entreprise a son siège à Paris mais possède une application ultra-utilisée par des clients à Singapour, répliquer les fichiers multimédias au plus près de ces utilisateurs réduit considérablement le temps de téléchargement.
-

2. Les deux types de Réplication

AWS sépare la réplication en fonction des frontières géographiques des infrastructures :

CRR (Cross-Region Replication)

- **Définition** : Source et destination sont dans des **Régions AWS différentes** (ex: `eu-west-3` Paris vers `us-east-1` N. Virginia).
- **Logique** : Répond aux besoins de **Disaster Recovery** mondial et de **réduction de latence** géographique.

SRR (Same-Region Replication)

- **Définition** : Source et destination sont dans la **même Région AWS**.

- **Logique** : Utilisé pour l'**agrégation de logs** (centraliser les logs de 50 buckets applicatifs d'une même région vers un seul bucket d'analyse) ou pour respecter les contraintes strictes de souveraineté des données qui interdisent le transfert transfrontalier.
-

3. Les Prérequis Techniques (Requirements) ?? *Point chaud de l'examen*

Si une question de l'examen mentionne que la réplication "ne fonctionne pas", la cause se trouve presque toujours dans le non-respect d'un de ces critères :

- **Le Versioning est obligatoire** : Il doit être explicitement activé sur le bucket source **ET** sur le bucket destination.
 - *Pourquoi ?* Sans versioning, si deux objets portent le même nom, l'un écraserait l'autre de manière asynchrone, créant des conflits de données impossibles à résoudre automatiquement pour S3.
 - **Le Rôle d'exécution IAM** : S3 doit recevoir l'autorisation d'agir en ton nom.
 - *Pourquoi ?* Par défaut, les services AWS sont isolés. S3 a besoin d'un rôle IAM pour obtenir le droit de lire (`s3:GetObjectVersion`) dans la source et d'écrire (`s3:ReplicateObject`) dans la destination.
 - **La Bucket Policy de destination (Cas multi-comptes)** : Si la destination est dans un compte AWS B, le propriétaire du compte B doit ajouter une politique de bucket autorisant explicitement le rôle IAM du compte A à déposer des objets.
-

4. Fonctionnement logique & Options Avancées

Comprendre le comportement par défaut de S3 permet d'éviter les pièges sur les options spécifiques.

Sans S3 Batch Replication (par défaut)

Par défaut, la réplication ne regarde pas le passé. Dès que la règle est activée, **seuls les nouveaux objets** (ou les nouvelles versions d'objets existants) sont répliqués.

Avec S3 Batch Replication

- **Pourquoi ?** Si tu as déjà 10 To de données historiques dans un bucket et que tu actives la réplication aujourd'hui, ces 10 To ne sont pas répliqués.
 - **Comment ?** Tu dois utiliser **S3 Batch Replication** (qui s'appuie sur S3 Batch Operations) pour scanner le bucket et répliquer de manière rétroactive les objets préexistants.
-