Les fondamentaux AWS - Partie III

RDS, Aurora et ElastiCache

AWS RDS - vue d’ensemble

- RDS: Relational Database Service
- Service géré de base de données relationelles utilisant le langage SQL pour les requêtes
- Permet de créer dans le cloud des bases de données maintenues par AWS
  - Posgres
  - MySQL
  - MariaDB
  - Oracle
  - Microsoft SQL Server
  - Aurora (propriété d’AWS)

Avantages par rapport à un déploiement EC2

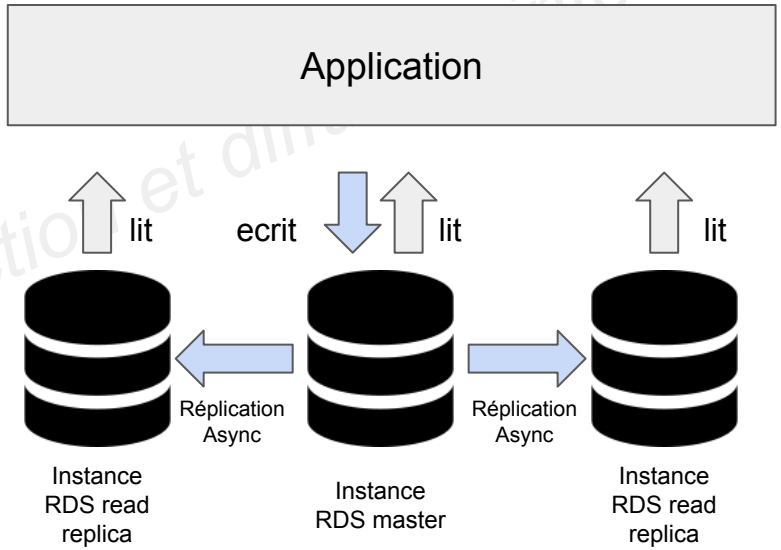
- RDS est un service géré
  - Approvisionnement automatique, correctif OS
  - Sauvegardes, restauration à un timestamp précis
  - Monitoring
  - Read replicas pour améliorer les performances
  - Multi AZ (Disaster Recovery)
  - Capacités de scaling (vertical et horizontal)
  - Stockage sur EBS (gp2 ou io 1)
- MAIS pas de connection via SSH

Sauvegardes RDS

- Automatiquement activées dans RDS
- Sauvegardes automatiques
  - Sauvegarde complète journalière
  - Logs de transaction sauvegardés toutes les 5 min
    - Possibilité de restaurer à un tout moment dans le temps (> 5 min)
  - Rétention de 7 jours (augmentable à 35)
- DB Snapshots
  - Déclenchés manuellement par l'utilisateur
  - Durée de rétention au choix

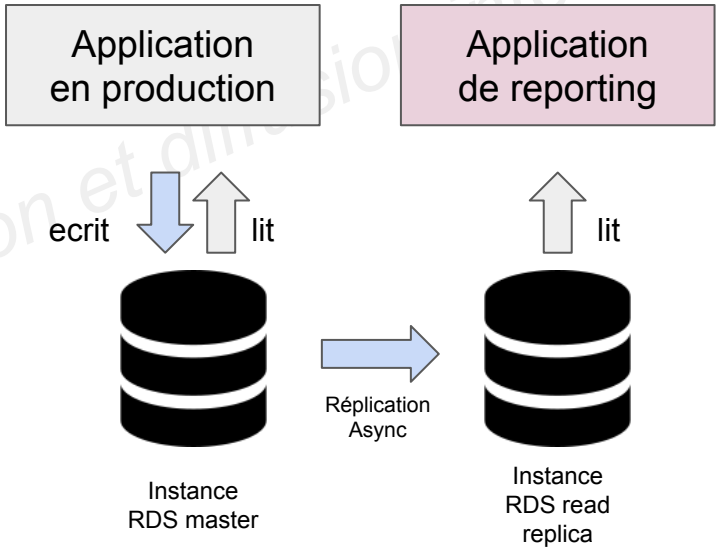
Replicas de lecture RDS

- Jusqu'à 5 Read Replicas
- Dans une seule AZ, multi AZ ou multi régions
- La réplication est ASYNC
- Les réplicas peuvent recevoir leur propre DB (master)
- Les applications doivent mettre à jour leur chaîne de connexion pour profiter des replicas de lecture



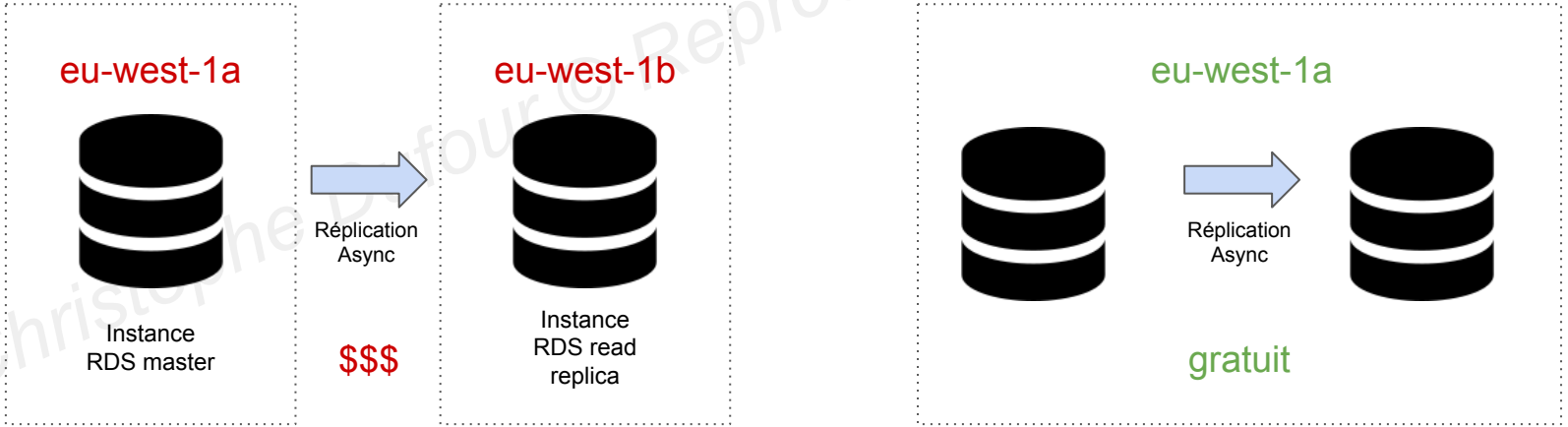
Replicas de lecture RDS - Cas d'utilisation

- Nous avons une base de données en production recevant une charge normale
- Nous voulons faire du reporting sur cette base
- Création d'un Read Replica pour cette nouvelle charge de travail
- L'application en production n'est pas affectée
- Les Read Replicas sont utilisés pour tout type d'opération SELECT (Et non INSERT, UPDATE, DELETE)



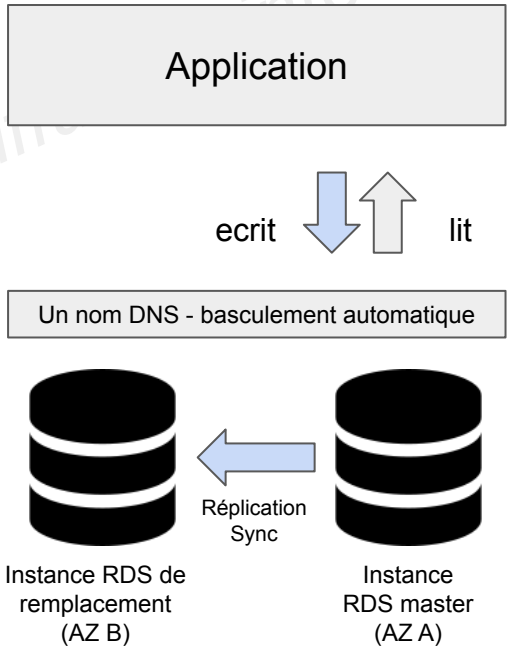
Replicas de lecture RDS - Coûts réseau

- Dans AWS, il y a des coûts réseau dès lors que des données vont d'une AZ à une autre
- Pour reduire ces coûts, on peut avoir des Read Replicas dans la même AZ



RDS Multi AZ (Disaster Recovery)

- Réplication SYNC
- Un nom DNS, basculement automatique vers une instance de remplacement
- Augmente la disponiblité
- Basculement en cas de perte d'une AZ, d'un réseau, d'une instance ou d'un échec de stockage
- Pas d'intervention manuelle dans les applications
- Ne s'utilise pas pour le dimensionnement



Sécurité RDS - Chiffrement des données

- Chiffrement au repos (at rest)
  - possibilité de chiffrer le master et les read replicas avec AWS KMS - AES-256 encryption
  - le chiffrement doit être défini au lancement
  - Si le master n'est pas chiffré, les read replicas ne peuvent pas être chiffrés
  - Transparent Data Encryption (TDE) disponible pour Oracle et SQL Server
- Chiffrement à la volée
  - Certificat SSL pour chiffrer les données vers RDS
  - Fournir les options SSL du certificat de confiance lors de la connection à la db
  - Pour forcer SSL
    - Postgres: rds.forceçssl=1 dans la console AWS RDS (Parameter Groups)
    - MySQL: GRANT USAGE ON *.* TO 'mysqluser'@'%' REQUIRE SSL;

RDS - opérations de chiffrement

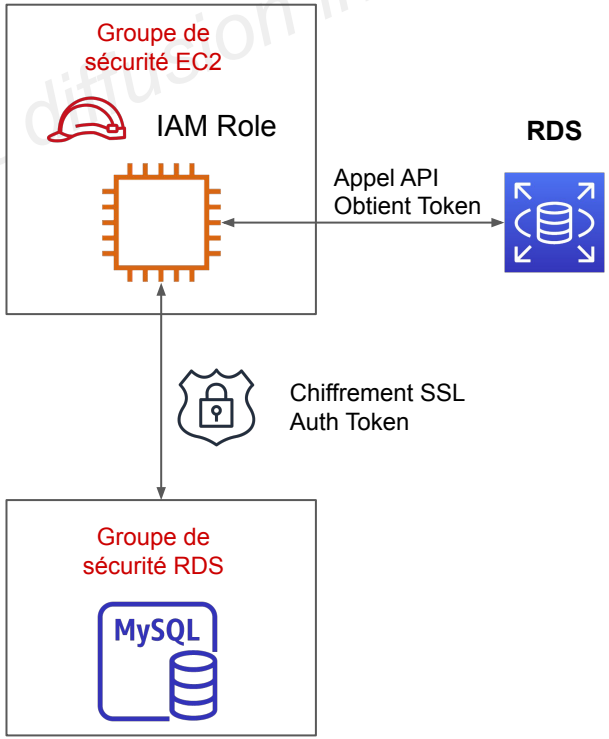
- Chiffrer des sauvegardes RDS
  - Les snapshots de bases RDS non chiffrées sont non chiffrés
  - Les snapshots de bases RDS chiffrées sont chiffrés
  - Possiblité de copier un snapshot dans une version chiffrée
- Pour chiffrer une base RDS non chiffrée
  - Faire un snapshot d’une base non chiffrée
  - Copier le snapshot et activer le chiffrement
  - Restaurer la base de données depuis le snapshot chiffré
  - Migrer les applications vers la nouvelle base de données et supprimer l’ancienne

Sécurité RDS - Réseau et IAM

- Sécurité réseau
  - Les bases de données sont généralement déployées dans un sous-réseau privé
  - La sécurité RDS s'appuie sur les groupes de sécurité (même concept que pour EC2), ils contrôlent quel IP / groupe de sécurité peut communiquer avec RDS
- Gestion d'accès
  - Les règles IAM aident à contrôler qui peut gérer AWS RDS (via RDS API)
  - Les traditionnels nom d'utilisateur / mot de passe peuvent être utilisés pour se connecter
  - L'authentification IAM peut être utilisée pour se connecter à RDS MySQL et PostgreSQL

RDS - Authentification IAM

- Fonctionne avec MySQL et PostgreSQL
- Pas besoin de mot de passe, juste un token d'identification obtenu via un appel à l'API IAM et RDS
- Le token d'authentification a une durée de vie de 15 minutes
- Avantages
  - Les entrées/sorties réseau peuvent être chiffrées via SSL
  - Gestion centralisée au niveau d'IAM plutôt que de la base
  - Permet d'exploiter les rôles IAM et les profiles d'instance EC2 pour une meilleure intégration



Sécurité RDS - résumé

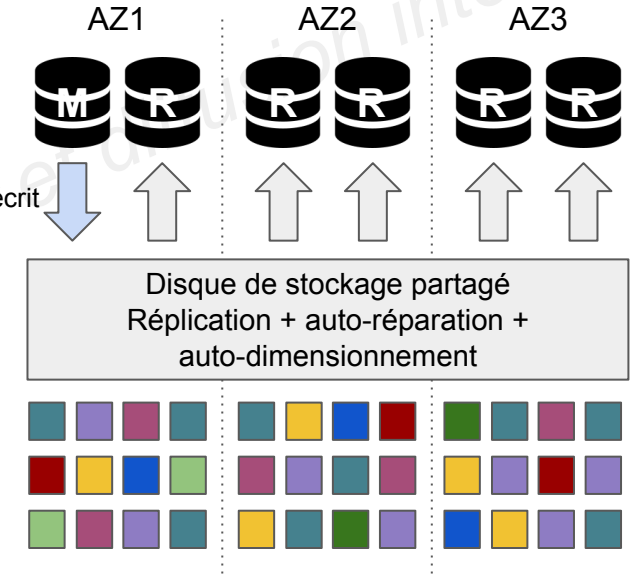
- Chiffrement au repos
  - effectué uniquement à la création de l'instance de base de données
  - sinon: db non chiffrée => snapshot => copie encryptée => création à partir de la copie
- Responsabilité utilisateur
  - vérification des IP/ports/règles en entrée des SG du SG de la base
  - création d'utilisateurs, au niveau de la base ou bien via IAM
  - création d'une base avec ou sans un accès public
  - vérification que les parameter groups ou la base est configurée pour n'accepter que des connections SSL
- Responsabilité AWS
  - Accès SSH
  - Correctifs appliqués à la base et à l'instance
  - Surveillance de l'instance sous-jacente

Amazon Aurora

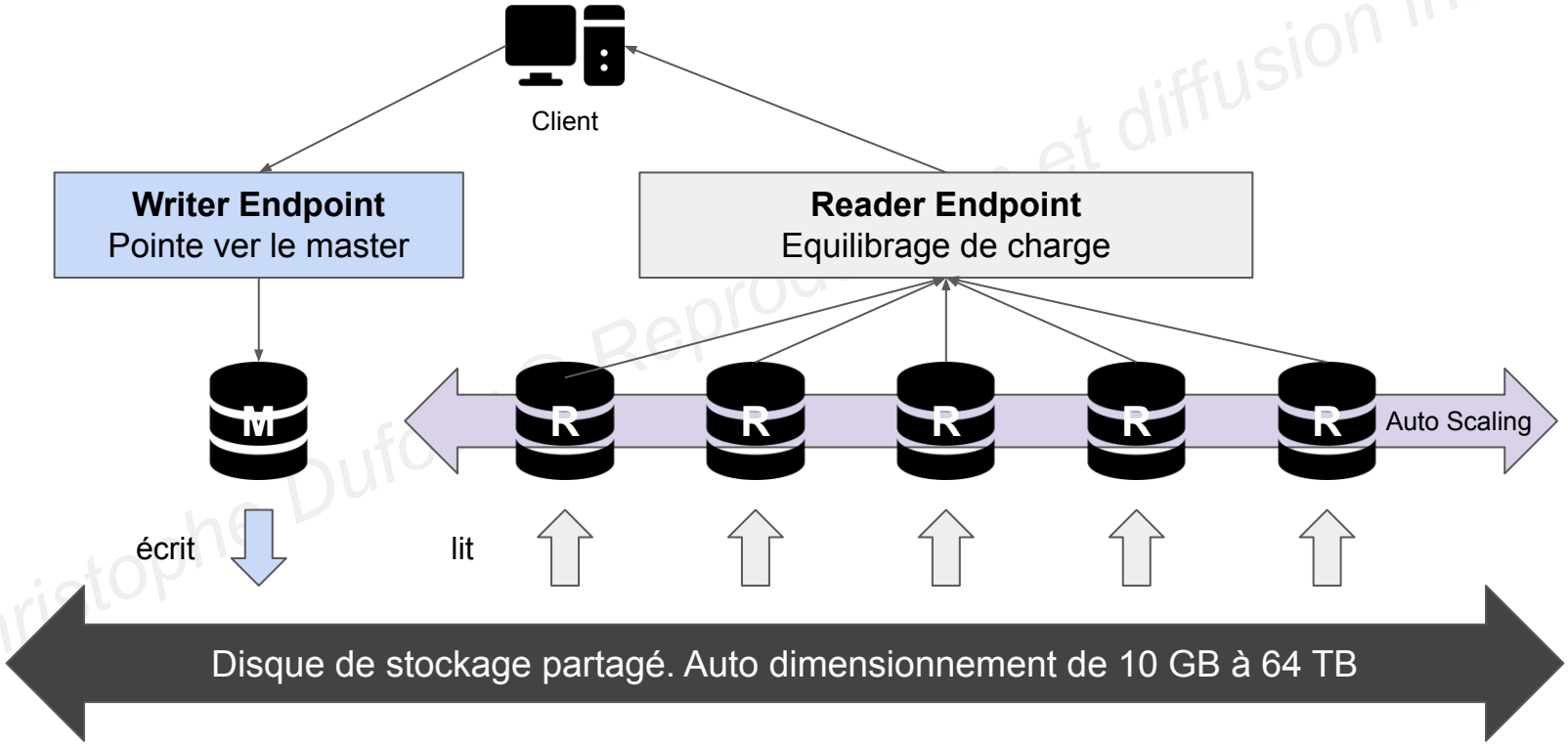
- Aurora est une technologie propriétaire développée par Amazon
- Postgres et MySQL sont tous deux supportés par Aurora (même drivers)
- Aurora est optimisé pour le cloud AWS et annonce des performances 5 fois supérieures à RDS MySQL 3 fois supérieures à RDS Postgres
- Le stockage augmente automatiquement jusqu'à 64TB par paliers de 10 GB
- Aurora peut avoir 15 replicas (contre 5 pour RDS) et la procédure de réplication est plus rapide (latence sous les 10 ms)
- Le basculement (failover) est instantané, il est nativement en haute dispo.
- Aurora est 20% plus coûteux que RDS mais plus efficace

Aurora - haute disponibilité

- 6 copies des données à travers 3 AZ
  - 4 copies sur 6 pour l'écriture
  - 3 copies sur 6 pour la lecture
  - auto réparation avec réplication peer-to-peer
  - le stockage est segmenté à travers des centaines de volumes
- Une instance Aurora (master) reçoit l'écriture
- Basculement automatisé pour le master en moins de 30 s.
- Master + jusqu'à 15 Read Replicas
- Support pour le réplications Multi régions



Aurora DB Cluster



Fonctionnalités d’Aurora

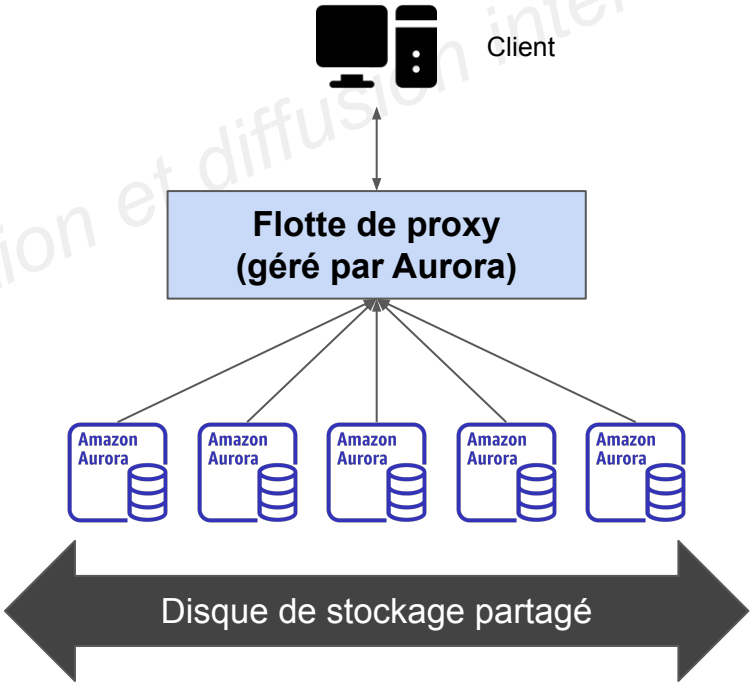
- Basculement automatique
- Sauvegarde et restauration
- Isolement et sécurité
- Conformité aux standards “industriels”
- Dimensionnement automatique
- Correctifs automatisés sans temps d’arrêt
- Monitoring avancé
- Routine de maintenance
- Retour en arrière: restauration de données à n’importe quel moment sans utilisation de sauvegardes

Aurora - sécurité

- Similaire à RDS
- Chiffrement au repos utilisant KMS
- Sauvegardes automatisés, snapshots et replicas chiffrés
- Chiffrement à la volée via SSL
- Possible authentification par token IAM (méthode identique à RDS)
- Utilisateur à la charge de protéger son instance avec les groupes de sécurité
- Pas de SSH

Aurora Serverless

- Instanciation de base de données automatisée et auto-dimensionnement basé sur l'utilisation
- Bonne solution pour des charges de travaille peu fréquentes, intermittentes ou imprévisibles
- Pas de plannification en capacité requise
- Facturation à la seconde, tarification peut être plus efficace



Global Aurora

- Réplicas de lecture Multi Region
  - Utile pour la récupération après désastre
  - Simple à mettre en place
- Aurora Global Database (recommandé)
  - 1 région primaire (écriture/lecture)
  - Jusqu'à 5 régions secondaires (lecture seule), temps de réplication < 1 s.
  - Jusqu'à 16 réplicas de lecture par région secondaire
  - Aide à réduire la latence
  - Promouvoir une autre région a une durée maximale d'interruption (RTO) < 1 min.