Engineer

Sauvegarde

une copie de données en cas de perte et sa création

Ce que c’est

Une sauvegarde est la copie à partir de laquelle les données sont restaurées lorsque l’original est supprimé, corrompu ou perdu avec la machine qui le détenait. Elle doit donc résider hors d’atteinte de ces événements. Le glossaire NIST la définit comme une copie de fichiers et de programmes destinée à faciliter la récupération. Le nom anglais backup, pluriel backups, désigne les copies, tandis que le verbe back up, écrit en deux mots, désigne l’action : une tâche nocturne sauvegarde la base et laisse une copie derrière elle.

Des éléments voisins sont souvent pris pour une sauvegarde. Une réplique reçoit chaque changement en quelques secondes : elle protège d’un disque mort et copie tout aussi fidèlement un `DELETE` erroné. Un instantané fige un volume à un moment donné et ne devient une sauvegarde que lorsqu’il est stocké là où les défaillances de l’original ne peuvent le suivre. Historique et séparation la définissent : retrouver les données telles qu’elles étaient auparavant, depuis un lieu ayant survécu à la perte de l’original.

Les bases rendent la distinction concrète. Le manuel PostgreSQL décrit trois approches fondamentalement différentes. Un export SQL avec `pg_dump` écrit les instructions reconstruisant la base, comme instantané cohérent du début de l’export. Une sauvegarde au niveau du système de fichiers copie le répertoire de données lui-même. L’archivage continu combine une telle sauvegarde de base avec le journal d’écriture anticipée (WAL) produit ensuite. Rejouer ce journal permet de restaurer tout instant depuis cette sauvegarde, capacité appelée restauration à un instant donné.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce qu'une sauvegarde ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Des sauvegardes testées plutôt que supposées fiables, et un plan écrit pour le matin où un serveur ne redémarre pas.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. 100 sauvegardes critiques automatisées : Barman — 100 sauvegardes critiques automatisées dans les bases

100 sauvegardes critiques automatisées : Barman Contactez-nous →

Pourquoi c’est important

Deux nombres transforment un souhait général de sécurité en conception. Le NIST définit l’objectif de point de reprise, RPO, comme l’instant auquel les données doivent être récupérées après une interruption : en pratique, le travail récent que l’entreprise peut se permettre de perdre. L’objectif de délai de reprise, RTO, est la durée pendant laquelle un système peut rester en récupération avant que l’interruption ne nuise à la mission de l’organisation.

Ces nombres agissent sur deux parties différentes. Le RPO dépend de la fréquence des copies : un export nocturne peut perdre une journée d’écritures, tandis que l’archivage WAL réduit la perte aux derniers instants de journal encore en transit vers l’archive. Le RTO dépend du temps de restauration. PostgreSQL doit rejouer chaque segment WAL depuis la dernière sauvegarde de base, ce qui peut être long si elle est ancienne. Sa fréquence relève donc autant du RTO que du stockage.

Les deux nombres sont des affirmations sur une restauration : c’est pourquoi une sauvegarde se juge en la restaurant. Une tâche sortant avec un statut zéro prouve que des octets ont été écrits quelque part. Pour savoir s’ils reconstruisent un système fonctionnel dans le RTO, il faut restaurer selon un calendrier, hors production, puis comparer ce qui revient.

La séparation est l’autre moitié. Une copie dans le même compte, sous les mêmes identifiants, partage les défaillances auxquelles elle doit survivre : compte fermé, clé volée, script exécuté contre la mauvaise cible. La question utile est l’événement unique qui emporterait cette copie avec l’original.

À quoi cela ressemble en pratique

À la plus petite échelle, la mise en œuvre des sauvegardes et tests de restauration de l'entreprise visait à étendre la reprise au-delà d'un instantané fournisseur. Un rôle Ansible installe scripts restic et unités systemd pour un export de base là où il est configuré, des instantanés chiffrés dédupliqués, la suppression selon la politique de conservation et les contrôles d'intégrité. Un processus mensuel distinct restaure un fichier connu dans un emplacement temporaire et compare ses octets. Sauvegarde et restauration ont des signaux de réussite séparés, et les temporisateurs nécessitent un dépôt configuré. La phrase secrète vient d'un fichier protégé plutôt que d'une valeur d'environnement interprétée par systemd. Ce sont des contrôles implémentés et des prérequis d'activation : le dossier de mise en œuvre du 2026-10-27 indique les temporisateurs désactivés sans dépôt et ne consigne aucune restauration réelle achevée. Il étaye le travail d'automatisation, pas une récupération réussie ni la possibilité de perdre déjà l'hôte sans risque.

À plus grande échelle, cent sauvegardes critiques ont été automatisées avec Barman et Google Cloud pour un produit dépendant de ses données SIG. Barman gérait PostgreSQL, Google Cloud conservait les copies hors site, et Bash et Python orchestraient et validaient les exécutions. Les contrôles d'intégrité confirmaient la validité de chaque sauvegarde plutôt que sa simple présence.

Sur un troisième produit, le travail consistait à définir la capacité de récupération plutôt qu'à la supposer. La conservation des sauvegardes réside dans le modèle Bicep à côté de la base. La restauration à un instant donné est ainsi une propriété du modèle plutôt qu'un réglage autrefois cliqué, et l'environnement alentour est lui aussi du code. L'audit a consigné les lacunes : géoredondance désactivée et RTO proposé plutôt que mesuré, avec l'exercice de mesure décrit mais restant à exécuter.

Où les problèmes surviennent

Les problèmes restent souvent invisibles jusqu'au besoin de restaurer. Chacun mérite donc d'être recherché un jour calme.

Une copie de fichiers d'une base active restaure une grappe défectueuse. Copier le répertoire PostgreSQL avec `tar` pendant son fonctionnement collecte des fichiers de moments différents. Le manuel exige l'arrêt du serveur pour une sauvegarde utilisable ainsi. Un instantané de système de fichiers pris

atomiquement fonctionne s'il inclut le WAL, rejoué au démarrage comme après une panne. Si données et WAL résident sur des systèmes de fichiers distincts, une prise simultanée peut être impossible. `pg_dump` ou une sauvegarde de base pour archivage continu donne une copie cohérente sans arrêter le serveur.

Une lacune dans l'archive WAL arrête la restauration à un instant donné à cet endroit. La récupération nécessite une suite ininterrompue de fichiers WAL archivés remontant au début de la sauvegarde de base. Si la commande d'archivage échoue continuellement, PostgreSQL réessaie et le WAL s'accumule dans `pg_wal/`. Le premier symptôme est donc souvent un disque qui se remplit. Le manuel conseille de configurer et tester l'archivage avant la première sauvegarde de base, puis d'en surveiller les retards.

La tâche s'arrête, et le silence ressemble au succès. Une tâche programmée défectueuse écrit son erreur dans un journal que personne ne lit, et chaque journée calme passe pour bonne. La supervision doit traiter le silence comme alarme : un signal envoyé uniquement après réussite à un service qui alerte en cas de retard détecte une tâche arrêtée aussi bien qu'un échec bruyant.

La clé disparaît ou arrive modifiée. L'accès aux données chiffrées dépend des clés. restic accepte plusieurs mots de passe pour un même dépôt ; son avertissement sur la perte irrécupérable des données s'applique lorsqu'il ne reste aucune clé utilisable pour ce dépôt. La clé doit résider dans un lieu survivant à la machine. Dans la configuration précédente, le changement aurait pu venir du superviseur lui-même : il interprète les échappements des variables d'environnement. Une phrase secrète contenant une barre oblique inverse serait donc arrivée à restic avec des octets différents de ceux de création du dépôt, sans aucun avertissement. La lecture depuis un fichier ferme cette voie.

La conservation supprime la copie nécessaire. Une corruption est souvent remarquée des semaines après. Une politique de sept copies quotidiennes a alors remplacé chaque copie saine par une version endommagée. Une politique à plusieurs niveaux conserve un historique long à faible coût, et `--dry-run` montre les suppressions prévues avant toute action :

```
restic forget --keep-daily 7 --keep-weekly 5 --keep-monthly 12 \
  --prune --dry-run
```

`forget` retire les instantanés et `prune` les données qu'eux seuls référençaient. Après ce nettoyage, les anciennes copies ont donc définitivement disparu.

Le contrôle d'intégrité passe, mais la restauration échoue. Sans option supplémentaire, `restic check` vérifie la structure du dépôt sans lire chaque fichier pack, car cela exige leur téléchargement complet. `--read-data` lit tout et `--read-data-subset=1/5` un cinquième : cinq passages de `1/5` à `5/5` couvrent l'ensemble. Une lecture complète ne prouve que l'intégrité des octets stockés. Seule une restauration montre s'ils reconstruisent une base fonctionnelle avec les bons rôles et réglages.

Pour aller plus loin

- Glossaire NIST : sauvegarde — définition utilisée ici et publications qui la fondent.
- NIST SP 800-34 Rev. 1 — guide de planification de continuité définissant RPO et RTO.
- PostgreSQL : sauvegarde et restauration — trois approches et coût de chacune.
- PostgreSQL : archivage continu et restauration à un instant donné — sauvegardes de base, archivage WAL et restauration à un moment choisi.
- restic : préparer un dépôt — mots de passe, fichiers de mots de passe et avertissement sur leur perte.
- restic : supprimer des instantanés de sauvegarde — conservation, `forget`, `prune` et `--dry-run`.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le parcours de déploiement, les risques opérationnels et les personnes d'astreinte. Nous pouvons définir un changement avec des vérifications, une procédure de reprise et une responsabilité claire.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce qu'une sauvegarde ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/backup/