



# Proxmox Backup Manager : Vérifier et Auditer Datastore



8 mai 2026



Mis à jour le 17 mai 2026



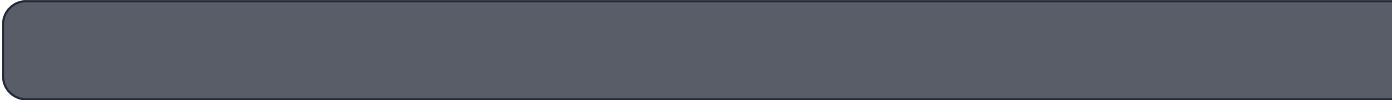
24 min de lecture



4949 mots



Vérifier l'intégrité d'un datastore Proxmox Backup Server (PBS) constitue un pilier souvent négligé de toute stratégie de sauvegarde robuste. La commande `proxmox-backup-manager datastore verify` garantit que chaque chunk dédupliqué correspond bien à son hash SHA-256 d'origine, détectant ainsi la corruption silencieuse (bit rot) et les défaillances matérielles. Ce guide explore le cycle complet : commandes CLI, audit du status, planification, gestion des erreurs, performance multi-thread, monitoring Prometheus/Grafana et chiffrement.



Vérifier l'intégrité d'un datastore Proxmox Backup Server (PBS) constitue un pilier négligé de toute stratégie de sauvegarde robuste. Au-delà du simple `backup` qui commande `proxmox-backup-manager datastore verify` garantit que chaque chunk dédupliqué stocké sur disque correspond bien à son hash SHA-256 d'origine, détectant ainsi la corruption silencieuse (bit rot), les défaillances matérielles et les erreurs de

Réponse sous 24h



de système de fichiers. Ce guide exhaustif explore l'ensemble du cycle de vérification d'un datastore PBS : commandes CLI complètes avec leurs options ( `--backup-id` , `since` , `--outdated-after` , `--filter` ), audit du status (espace, déduplication, garbage collection), planification automatique via jobs PBS et cron Linux, gestion des erreurs (corruption, optimisation des performances multi-thread, intégration au monitoring Prometheus/Grafana, gestion du chiffrement et stratégies de récupération. Vous trouverez également trois scénarios concrets (disque défaillant, après upgrade majeur, post-migration) ainsi qu'une FAQ détaillée. Cet article s'adresse aux administrateurs Proxmox confirmés gérant des environnements de production où la fiabilité des sauvegardes n'est pas négociable.

#### À RETENIR

### À retenir

**Verify = intégrité cryptographique** : la commande `proxmox-backup-manager datastore verify` recalcule le SHA-256 de chaque chunk et le compare avec le manifest. Indispensable pour détecter le bit rot.

**Status = audit capacitaire** : `datastore status` donne taux de déduplication, espace utilisé, dernier GC. Complémentaire à verify, pas substituable.

**Planification obligatoire** : configurer un Verify Job hebdomadaire dans la console PBS ou via cron Linux. Ne jamais se contenter de vérifications manuelles ponctuelles.

**Performance** : verify est I/O bound (lecture intégrale du datastore). Prévoir une fenêtre maintenance, utiliser `--ignore-verified` et `--outdated-after` pour réduire la charge.

---

---

Réponse sous 24h

Devis  
gratuit

