Rehearse a supported CoreProtect database-backend migration

CoreProtect’s migration command is available only in specific supported builds and backends. Rehearse the matching procedure before changing where a live server stores its audit history.

Follow the procedure

1. Record the installed CoreProtect build and current backend. Check the migration documentation for feature availability, supported destination types and any release or licensing requirement.
2. Take a consistent backup of the database and plugin configuration. Preserve a matching world recovery copy if the audit history is part of an ongoing restoration workflow.
3. Prepare an isolated server copy and a new destination namespace with no existing CoreProtect data. Keep the source database and production connection untouched during rehearsal.
4. Follow the documented console-only migration workflow for that release. Review its validation mode and ensure the required database-lock setting remains in the documented state.
5. Compare known historical records and active logging after migration. Where supported, choose full validation when the rehearsal’s time budget permits rather than assuming sampled validation checks every row.
6. Record duration, disk requirements and the configuration changes made by the migration tool. Plan a controlled production window only after the rehearsal has completed and its data checks pass.

   CloudExa game panel: create a server backup
   CloudExa game panel reference. This image shows the panel, not the CoreProtect interface.

Check the result

The isolated destination should contain the expected records and accept new test actions through a clean plugin startup. Verify several old locations and a new action, plus the migration’s reported validation result. Successful connectivity alone is insufficient evidence of a complete history transfer.

If the result is wrong

If validation fails or the installed build lacks the feature, keep production on its current backend and retain the rehearsal logs for support. Do not bypass checks by copying guessed tables. Roll back the matched database and configuration rather than swapping only a jar across schema changes.

References and related help

- CoreProtect maintainer reference
- CoreProtect maintainer reference
- Inspect block changes and preview a CoreProtect rollback
- Recover a Minecraft server after griefing

Revision #1
Created 2026-11-09 16:36:18 UTC by CloudExa Hosting Admin
Updated 2026-11-09 16:36:18 UTC by CloudExa Hosting Admin