# Inspect block changes and preview a CoreProtect rollback

CoreProtect records supported actions while its logging is enabled. It is useful for investigating who changed a block and reversing a specific recorded incident. It cannot recreate history from before installation or after the relevant records have been purged. Keep ordinary server backups as a separate recovery method.

# Check logging before an incident

Install a compatible release and inspect the generated configuration for the events you need recorded. Confirm the database is working and has room to grow. On a test area, place and break a harmless block with a known account, then check that the history appears.

Give investigators only the permissions they need. Inspection and lookup are different capabilities from rollback, restore and purge. Keep database access and destructive administrative commands restricted. A staff title or a panel subuser account does not automatically define the correct Minecraft permissions.

# Inspect the affected location

In game, enable the inspector with `/co inspect` and examine the affected blocks using the documented interactions. Run the command again to turn inspection off when finished. Record the world, coordinates, suspected player and approximate time rather than guessing a server-wide recovery period.

Use a lookup before changing anything. An example from the affected location is `/co lookup u:PlayerName t:30m r:10 a:block`. Replace the player deliberately. This searches recorded block actions within a ten-block radius during the last thirty minutes. Run it in game at the correct location; a radius is relative to where you execute it.

If the result is empty, check the account, world, time window and logging configuration. No result does not prove the player did nothing, especially when logging began after the incident.

# Back up, preview and narrow the scope

Before applying a rollback, stop player changes in the area and take a completed server backup. Preserve the matching CoreProtect database and any relevant external plugin data. Test the proposed operation on an isolated copy for valuable or complicated incidents.

Use the same confirmed filters for a preview, for example `/co rollback u:PlayerName t:30m r:10 a:block #preview`. Review the preview and its confirmation instructions carefully. Previewing is an opportunity to reject an overly broad selection; it is not a reason to skip the backup.

Restrict the action, time, player and location to the incident. Do not substitute a global radius because a local lookup missed something. Avoid including container or inventory actions unless you understand how those recorded events interact with the recovery you need.

# Apply only the reviewed operation

Proceed using the plugin's displayed confirmation flow only after the preview matches the intended repair. Keep players out until processing completes. Inspect repaired blocks, nearby builds and relevant containers before reopening the area.

A CoreProtect restore can reverse recorded rollback effects, but it is not a promise to recover every later interaction or external plugin state. Keep the pre-operation backup and command record. Do not purge the database during an investigation. If the scope remains uncertain, send support redacted lookup output, coordinates and the incident time rather than experimenting with a larger rollback.

# Related CloudExa guides

- Recover a Minecraft server after griefing
- Back up game files before making changes

# Further reading

- CoreProtect command reference
- CoreProtect permissions
- CoreProtect configuration

---

Revision #2
Created 2026-11-08 21:45:13 UTC by CloudExa Hosting Admin
Updated 2026-11-08 21:45:13 UTC by CloudExa Hosting Admin