Keep default-player permissions separate from staff inheritance

A default group shared by every player should contain only ordinary-player access. Audit it before moving common commands into a convenient but overpowered staff parent.

Follow the procedure

1. Choose two test accounts: an ordinary player and a limited staff member. Record their current parent groups and a small list of actions each should perform.
2. Inspect the default group’s direct permissions and parents. Look for moderation, bypass, administrative or broad wildcard nodes that ordinary players do not need.
3. For each suspicious node, identify the plugin feature it controls and who actually needs it. Do not remove unknown nodes without checking their documented behaviour.
4. Move one staff-only grant into the appropriate staff group during a controlled change, preserving its old value and contexts in the change record.
5. Test the ordinary account for denial and the staff account for allowed access. Also confirm ordinary commands still work for both accounts through the intended inheritance.
6. Continue in small groups of related nodes. Keep an explicit list of the base player features so future administrators can add permissions without using a staff group as a shortcut.

Check the result

Ordinary players should retain useful player commands and lose staff-only features. Staff should receive those features from their reviewed role. Test an actual action such as a harmless read-only staff command, not an operator account whose default permission behaviour can hide mistakes.

If the result is wrong

If a common player action fails, restore only that action’s required node to the base role after checking documentation. Avoid restoring all removed grants at once, because that also restores the unsafe ones. Use the recorded baseline to distinguish an intended denial from an accidental loss of normal access.

References and related help

- LuckPerms maintainer reference
- LuckPerms maintainer reference
- LuckPerms maintainer reference
- Create Minecraft ranks and inheritance with LuckPerms
- Trace a denied Minecraft command with LuckPerms verbose

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