Plan a Rust wipe and choose a map seed safely

A Rust wipe is a planned reset, not a repair for every startup problem. Decide which progress should disappear before touching files. A seed change alone does not describe what happens to blueprints, plugin economies or player records.

Choose the reset you actually want

- **Map wipe:** replace the active world. Terrain, placed structures and world entities belong to that world; describe the effect on inventories explicitly in your announcement.
- **Player-data reset:** reset the relevant player state. This is a separate decision from selecting a new map, and plugins can maintain their own records.
- **Blueprint wipe:** remove learned crafting blueprints. A community can keep blueprints across a map wipe, unless a game update or its own policy requires otherwise.
- **Full community reset:** agree whether plugin balances, kits, teams, statistics and permissions are included. They are not necessarily stored with the map.

Record and protect the current setup

1. Tell players the maintenance time, timezone and exact reset scope. Check Facepunch's current update announcement rather than assuming every reset follows an old calendar.
2. Record the current startup settings, server identity, map type, seed and world size. Include any custom-map URL and framework or plugin versions.
3. Stop the server cleanly through the panel and wait for it to be fully offline. Make a completed backup and download a separate copy. Include the identity directory, configuration and plugin data. Confirm you can browse the archive before continuing.

Apply the intended map change

For a procedural world, Rust uses `server.level`, `server.seed` and `server.worldsize`. Keep the level set to `Procedural Map` when using a procedural seed. A familiar seed is not a promise of identical terrain after future game-generation changes.

Use the configuration exposed for your service. Do not replace its complete launch command with an example from another host: that can change ports or the active save identity. If the relevant setting is unavailable, ask support to apply the recorded change.

For the agreed wipe, identify the active world and any separately selected player or blueprint records before moving them out of the active location into your protected archive. Do not run wildcard deletion commands. If you cannot distinguish the files, stop and ask support to identify them. Preserve unrelated saves and plugin databases.

A custom map instead uses `server.levelurl` and a direct map download. Do not combine conflicting custom-map and procedural settings. Changing an existing map's terrain underneath buildings is not a safe way to preserve a wipe's progress.

Check before reopening

Start once, watch generation complete, join and inspect the expected map. Check blueprint retention and representative plugin records against your announcement. A wipe countdown or weekly tag does not prove an automated reset is configured. Keep the pre-wipe backup until the new world has been accepted.

Further reading

- Facepunch: creating a Rust server
- Facepunch: server wipe timer
- Facepunch: hosting a custom map

Revision #2
Created 2026-11-08 21:28:53 UTC by CloudExa Hosting Admin
Updated 2026-11-08 21:28:53 UTC by CloudExa Hosting Admin