Investigate and recover from a full game server disk

A full disk can prevent Minecraft from saving worlds, writing logs or completing a backup. Treat the first storage warning as a reason to inspect usage. Adding RAM will not increase your storage allowance.

Protect current progress

If the game still responds, arrange a pause and stop it through **Console**. Wait for the shutdown result and note any save error. Avoid repeated restarts, imports or restores while storage is exhausted; they can create more temporary files or make recovery harder.

Write down your server ID, current disk usage and limit, and when the problem began. Read the resource usage guide if you need to distinguish the disk figure from memory.

Find likely growth

> Inspect Files before removing anything. The example shows world, plugin and log folders; it dc
> Inspect Files before removing anything. The example shows world, plugin and log folders; it does not identify files safe to delete. Select the image to enlarge it.

1. Open **Files** and inspect the server's top-level directory.
2. Look for uploaded ZIP or tar archives, especially after an import. An archive and its extracted contents can both remain on disk.
3. Check known world folders, including additional dimensions and worlds created by plugins.
4. Inspect logs, crash reports, rendered maps and plugin data. Record unusually large files and repeated errors rather than deleting their directories.
5. Check whether a backup, restore, upload or import is still active. Ask support before altering files involved in an active operation.

A file manager may show individual file sizes without calculating every directory's total. If you cannot account for the panel's figure, ask support for a read-only usage breakdown. Hidden files,

container bookkeeping or temporary operations may need a node-side check.

Reclaim only space you can account for

Start with an old upload archive that you no longer need on the server. Download it, verify that the download opens and contains the expected files, then keep that independent copy before choosing to remove the server copy. A matching filename alone does not prove the copy is complete.

Preserve worlds, player data, plugin databases, permissions and configuration. Do not remove a folder because its name looks unfamiliar. Do not delete the last known working recovery point or erase a world to make an update start.

If there is not enough space to create another local archive, download existing files through the available file/SFTP tools or ask support about a safe export path. Creating a large new archive on an already full disk can fail. External MySQL data requires its own export.

Check the cause before starting again

When logs grow quickly, save the relevant error and fix its source with the plugin developer or support. Deleting the log alone leaves the failure in place. World exploration, pre-generation and map rendering can also use substantial space; review their configuration before resuming them.

Wait for the panel's usage figure to refresh after a confirmed cleanup. Keep room for normal saves, updates and temporary operations. Start the server, check the console for save errors, then join to verify the expected world and player data.

If the usage still exceeds what you can explain, send support the limit, reported usage, operation in progress and the largest files you found. Request a storage upgrade if normal retained data genuinely needs more capacity. Specify storage separately from RAM and ask for a resize that preserves the installation.

Next: backup and restore checks or collect useful error logs.

Revision #2
Created 2026-11-08 20:06:28 UTC by CloudExa Hosting Admin
Updated 2026-11-08 20:06:28 UTC by CloudExa Hosting Admin