Choose tModLoader mods or TShock plugins for a Terraria server

Terraria gameplay mods and administration plugins use different server software. Decide which type of change you need before uploading files. A package downloaded for one ecosystem will not become compatible simply because it is placed in another server's directory.

Choose the correct software family

- **Vanilla Terraria:** the standard server for the base game. It does not load arbitrary tModLoader packages or TShock plugins.
- **tModLoader:** use this for its supported gameplay mods and corresponding clients. The upstream usage guide explains its mod workflow and separate save locations. A vanilla client is not a substitute for the required tModLoader client.
- **TShock:** a server administration and plugin ecosystem built on Terraria Server API. The upstream API project loads server plugins from `ServerPlugins`. These DLL plugins are not `.tmod` gameplay packages. Check each plugin's supported Terraria, TShock and API releases.

Do not attempt a combined TShock/tModLoader installation unless a separately supported project explicitly provides that arrangement. CloudExa cannot infer compatibility from similar feature names.

Prepare and install one compatible change

1. **Inventory the live service.** Record the executable family, exact release, startup configuration, selected world and active save paths. Identify whether any server-side character or account database is in use.

2. **Make a clean backup.** Save and stop normally, then preserve the complete world, player-related data, configuration, mods/plugins and any plugin databases. Keep both files of a tModLoader world, `.wld` and `.twld`, together; preserve complete player saves and their sidecar data too.
3. **Use an isolated test copy.** Select a package from its author's release page for your exact software family. Read dependency and migration notes before copying it. Keep credentials and production database connections out of the test environment.
4. **For tModLoader, transfer the intended enabled set.** Its current Dedicated Server Utils documentation describes exporting a mod pack and using `install.txt` plus `enabled.json`. Those instructions apply to that upstream layout; a hosted template may use different active paths. Confirm the service's configured mod directory instead of uploading to a guessed location.
5. **For TShock, follow the plugin's release instructions.** Place the compatible plugin in the active server's `ServerPlugins` directory, with documented dependencies. Keep existing TShock components intact. Grant only the permissions the feature needs.

Check the feature and the existing data

Start the test and confirm the expected mod or plugin loads without dependency errors. For tModLoader, join with the matching branch and required mod set. For TShock, test a normal account as well as an administrator; unrestricted administrator access can hide permission mistakes.

Inspect an existing structure, inventory and the new feature, then save and restart. If the test fails, preserve its logs and restore the complete recorded combination on the test copy. Removing a content mod and saving the only live world is not a safe rollback. Schedule the live change only after the test passes, and retain the previous files.

Related: upload a Terraria world and grant limited TShock permissions.

Revision #1
Created 2026-11-10 11:05:47 UTC by CloudExa Hosting Admin
Updated 2026-11-10 11:05:47 UTC by CloudExa Hosting Admin