Configure an ExcellentCrates test crate and verify rewards

ExcellentCrates manages crates, keys and rewards. Build a private test crate before distributing keys or announcing prizes. A successful animation is not proof that the correct item, balance or permission was awarded, and a mistaken reward command can affect more than the person opening the crate.

Use documentation for the installed major version

Check the release's supported Minecraft, Java and server platforms and install its required NightCore framework. Optional integrations should be added only for features you use. ExcellentCrates has changed its crate, key and block workflows across major versions; do not paste an older tutorial's commands into a newer installation without checking the command help.

Follow plugin installation, stop normally and keep a backup of configuration, crate definitions, player/key data and any external database. Preserve the previous JAR and matching framework version. A database format conversion may make downgrading unsafe even when the old JAR is still available.

Create an isolated reward test

1. Start and inspect Console for required-plugin or loading errors.
2. On the documented v7 command family, use `/ecrates help` and `/ecrates status` to inspect commands and active modules.
3. Open the documented crates editor using `/crates editor`. Create a distinctly named test crate rather than modifying a crate already in public use.
4. Add one small, unmistakable item reward through the editor. Read its quantity, chance or weight, eligibility and execution settings before saving.
5. Use the matching key editor and current key command help to create and issue a limited test key to your own test account.

Aliases can be changed in configuration. If the installed command differs, inspect its help rather than repeatedly trying commands from another release. Keep the test inaccessible to ordinary players until the whole transaction has been checked.

Verify keys, opening and delivery

Test a valid key and an invalid or missing key. Count keys before and after one opening, then inspect the actual reward received. Try a full inventory to understand the plugin's configured delivery behaviour. Verify that quitting or reconnecting does not grant a second reward from the same completed opening.

For physical crate blocks, follow the workflow for your exact release. Older block-linking instructions do not necessarily apply to newer modular crate-block systems. Check that placement occurs in the intended world and that region protection allows the interaction without allowing visitors to edit the crate.

Review commands before adding valuable prizes

A console reward executes with server authority. Use the documented recipient placeholder, restrict editors to trusted staff, and avoid operator grants or commands built from arbitrary player input. Test money rewards against the economy provider and check balances independently. See economy integration.

Back up before deleting a crate or changing its reward structure. Some releases remove linked rewards when a crate is deleted. For failures, preserve key counts and transaction logs, then report the crate ID, test player, timestamp, versions and redacted error rather than clearing player data.

Further reading: requirements, commands and workflow changes.

Looking for another plugin? Browse the Minecraft plugin guide directory.

Revision #2
Created 2026-11-09 08:57:17 UTC by CloudExa Hosting Admin
Updated 2026-11-09 08:57:17 UTC by CloudExa Hosting Admin