Test InteractiveChat item previews without exposing staff inventory

InteractiveChat can share a held item, an inventory or an ender chest through chat. Start with the held-item feature and two disposable accounts. Staff inventories can contain private tools or named items, so use test items rather than a real moderator's inventory for screenshots and previews.

Check dependencies and chat ownership

Choose an InteractiveChat release that supports your Minecraft version and chat platform. The project lists dependencies including ProtocolLib, PlaceholderAPI and Vault. Verify the requirements of the exact download and confirm those plugins load before diagnosing message formatting. A proxy installation needs its own documented network setup.

Record which plugin formats chat, whether a proxy relays it, and whether Discord receives copies. Preserve InteractiveChat and chat-plugin settings before editing. Keep tokens and private message content out of support captures. A successful local preview does not establish that a relay handles the same component correctly.

Limit the first feature

1. Open the configuration generated by your installed release. In the documented `ItemDisplay` section, inspect the separate `Item`, `Inventory` and `EnderChest` options.
2. For this trial, leave held-item sharing enabled and disable whole-inventory and ender-chest sharing. Record the previous values so you can restore your intended policy later.
3. Review the held-item keyword and a modest cooldown. In the current reference configuration, the default keyword recognises `[item]` and `[i]`.

4. Apply changes using the release's documented reload procedure or a normal restart, then check startup output for configuration errors.

Run a two-player preview

Join as a normal test player holding a disposable named item. Send a short public message containing `[item]`. Have a second normal player inspect the tooltip and, if your configuration allows it, open the read-only preview. Confirm the item name, quantity and lore match the shared test item.

Try an empty hand and a second message during the configured cooldown. Check that neither case floods chat or produces a stack trace. Attempt to move an item from the preview using disposable contents and verify that the viewer gains no inventory items. Do this on a private copy before enabling unfamiliar view integrations.

Review information exposure

Custom lore, map previews and compass metadata can reveal more than the base material. Current configuration includes a setting for hiding lodestone compass position. Review its comments if your server uses tracking compasses. Decide whether you want clickable maps and container previews before enabling those features.

Test private-message commands and external relays separately. The plugin can parse configured commands as well as public chat. Keep staff-only channels out of the trial until you have checked their audience and permissions. An addon may be required for a particular relay; do not assume the local preview transfers unchanged.

Recover from formatting conflicts

If a keyword remains literal, first check feature enablement, effective permissions and the chat formatter's compatibility notes. If a preview breaks only across a proxy, preserve the working local configuration while investigating the relay. Restore the previous InteractiveChat settings to reverse the trial; avoid disabling authentication or granting broad staff access as a workaround.

Related CloudExa guides

back up the server files · review LuckPerms groups and inheritance · collect the relevant Minecraft logs

Official references

- InteractiveChat features and dependencies
- InteractiveChat generated configuration reference

Revision #1
Created 2026-11-09 10:42:41 UTC by CloudExa Hosting Admin
Updated 2026-11-09 10:42:41 UTC by CloudExa Hosting Admin