DeluxeMenus tasks

Task guides and troubleshooting for DeluxeMenus. Check the installed release before changing configuration or data.

- Restrict one DeluxeMenus button without hiding the entire menu
- Choose player or console execution for a DeluxeMenus action
- Fix a DeluxeMenus item missing from a specific slot
- Prevent a DeluxeMenus open command from conflicting with another plugin
- Trace a DeluxeMenus placeholder before changing a purchase requirement
- Check left and right click actions separately in DeluxeMenus
- Tune a DeluxeMenus refresh interval for changing display values
- Repair one DeluxeMenus YAML edit while preserving the working menu

Restrict one DeluxeMenus button without hiding the entire menu

A player may need to see an upgrade button while lacking permission to use it. Decide whether your rule controls visibility, execution or both before adding requirements to the menu.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Copy the menu and identify the item’s unique configuration key and slot. Keep the existing open command available so a failed test does not remove the normal player route.
2. Choose the documented view requirement when the item should disappear, or a click requirement when it should stay visible but refuse the action. Avoid placing this rule on the entire menu.
3. Attach a clear denial message to the rejected action and keep the click command defined. The requirement guide notes that success commands alone do not replace the associated click command path.
4. Test an allowed account and a denied account with the same item and click type. Include alternate clicks that the item defines so another action cannot bypass the intended restriction.

Check the result

The item should appear and respond according to the written rule for both accounts. Confirm that nearby unrestricted buttons still work and that a denied click does not consume money or items.

If the test fails

Restore the copied item’s original requirements if unrelated buttons disappear or legitimate actions fail. Correct the requirement level before touching the permission provider or granting a broad menu bypass.

Keep access checks on the action that spends currency or grants a reward. Visibility alone does not establish that an executable action is authorised.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Create Minecraft ranks and inheritance with LuckPerms

Choose player or console execution for a DeluxeMenus action

A button can work for staff but fail for players because the command runs as its clicker. Switching everything to console can conceal the problem while granting far more authority than intended.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Identify the exact command and its owner. Test that command as an ordinary account outside the menu and record the minimum permissions and arguments it requires.
2. Check the action prefix in the copied item. The documented player action uses the clicker’s permissions, while the console action uses server authority; choose the identity the feature actually needs.
3. Keep player-controlled arguments out of privileged command strings. Use a fixed test action first and review whether any supplied player name or value could change the command’s meaning.
4. Test the copied button with an allowed and a denied account. Include a malformed or missing argument where the menu accepts input, and verify that rejection happens before the privileged action.

Check the result

The intended account should receive only the planned result. Inspect the relevant plugin state or audit record as well as the menu message, since a success message alone cannot prove correct

execution.

If the test fails

Revert the action prefix and temporarily close the copied route if a denied account can trigger the command. Remove any unintended test grant through its owning plugin and keep the menu evidence for review.

A console action deserves a fixed, permission-checked workflow. Do not turn a general command launcher into a player-facing console bridge.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Find Minecraft logs and send useful troubleshooting details

Fix a DeluxeMenus item missing from a specific slot

A blank slot can result from a hidden item rather than an invalid material. Isolate the slot and the viewer conditions before changing the menu size or replacing its whole item section.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Copy the menu and record its inventory type, size and target slot. Compare those values with the installed release’s GUI and item documentation rather than guessing how a different inventory numbers slots.
2. List every configured item that can occupy that slot. Check each item’s view requirements and priority so you can identify which version should appear for the test account.
3. Replace only the test item’s display with a simple supported material and fixed label. Leave rewarding click actions disabled in this copy while deciding whether the item renders.
4. Open the copy as an ordinary player and then as the alternate rank. Add placeholders and conditional versions back one at a time, checking the slot after each supported refresh.

Check the result

The slot should show the intended item for each viewer condition without covering a different button. Test all configured menu pages that reuse the same slot number before approving the change.

If the test fails

Restore the original item section if the simplified test still fails and inspect the first menu parsing error. Keep the working public menu intact; deleting other items can hide the collision without explaining it.

A conditionally hidden item can be correct behaviour. Write down which account should see which version so future changes preserve the intended selection.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Find Minecraft logs and send useful troubleshooting details

Prevent a DeluxeMenus open command from conflicting with another plugin

A menu may open through an administrative route but fail through its advertised command. Compare command ownership before changing the menu’s item actions or giving players extra administrative permissions.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Check the menu’s configured open command and list other plugins using the same name. Record whether the command fails, opens another feature or displays a permission message.
2. On a copied menu, choose one unused single-word open command as described in the GUI documentation. Preserve the original route until the new command has a clear owner.
3. Use the plugin’s supported configuration refresh procedure and inspect its command registration messages. Test the new route as an ordinary player, not through an administrator’s open-for-player command.
4. Check aliases, chat links and NPC or menu actions that advertise the old route. Update only the references you own after the new command opens the correct copied menu.

Check the result

The new command should reach the intended menu without invoking another plugin. Confirm that both allowed and denied players receive the intended access result and that unrelated commands retain their previous behaviour.

If the test fails

If registration fails, restore the previous open command and collect the startup or refresh error. Do not disable a working unrelated plugin simply to free a command name; choose a documented non-conflicting route.

Command names, permissions and menu identifiers serve different purposes. Changing one does not automatically update references to the others.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Find Minecraft logs and send useful troubleshooting details

Trace a DeluxeMenus placeholder before changing a purchase requirement

A purchase denial can result from raw text where the requirement expects a number. Trace the displayed and compared values before adjusting prices or adding money to a player’s account.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Copy the purchase item and record its price, requirement type, compared expression and provider. Keep the rewarding action disabled in the diagnostic copy to avoid accidental test purchases.
2. Parse the expression directly for a connected test player using the provider’s supported diagnostic route. Check whether the result contains currency symbols, separators or an empty string instead of a plain numeric value.
3. Compare the requirement with the installed DeluxeMenus documentation. Choose a documented comparator and a provider value with the expected type; do not silently convert an unknown string into zero.
4. Test balances below, equal to and above the price on disposable accounts. Re-enable the test action only after the requirement and denial message agree with those boundary cases.

Check the result

The purchase should reject insufficient funds and accept the intended boundary value without granting a reward twice. Confirm the account balance and delivered result independently of the menu’s message.

If the test fails

Restore the previous requirement if parsing remains uncertain and close the test purchase route. Do not compensate by granting bypass permissions or editing production balances; fix the provider or expression first.

Keep the displayed price and the enforced requirement in the same test notes. A correct label can still advertise a different price from the action.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Find Minecraft logs and send useful troubleshooting details

Check left and right click actions separately in DeluxeMenus

A button can define more than one click action. Testing only a normal left click can miss a right-click path that spends money, sends a command or ignores a requirement.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Copy the item definition and list its configured click command and requirement sections. Include shift-click variants and any fallback action documented for the installed release.
2. Write the intended result for each defined click type. Leave undefined actions harmless and ensure that a requirement belongs to the corresponding command path rather than a different click section.
3. Test each action with an ordinary allowed account and a denied account. Use disposable balances or items when the button changes state, and inspect the owning plugin’s records after each click.
4. Repeat a short sequence of different clicks without closing the copied menu. Check whether a result updates immediately and whether a second click can repeat an action that should no longer be available.

Check the result

Each defined click should produce its own intended result and enforce the same required policy. Compare before-and-after balances, inventory or permissions rather than counting displayed

success messages.

If the test fails

Restore the item’s previous action sections if a path bypasses the rule, and withdraw the copied menu’s opening route while fixing it. Reverse only the disposable test changes through their owning plugins.

Hide or disable a risky action until its state change is understood. A player’s normal clicking pattern is not a reliable access-control boundary.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Create Minecraft ranks and inheritance with LuckPerms

Tune a DeluxeMenus refresh interval for changing display values

A menu can show stale values because its refresh interval is too slow for the information you display. A faster interval can also repeat unnecessary provider work across many viewers.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Record the copied menu’s update interval and all values that change while it stays open. Separate a live countdown from a rank label that rarely changes during a session.
2. Check the current GUI documentation for the update control and its unit before editing it. Keep a screenshot and saved configuration of the existing behaviour for comparison.
3. Change the interval for the one copied screen, then alter a harmless test value while an ordinary account keeps the menu open. Measure the visible delay instead of judging an unchanged display.
4. Compare server performance under a similar number of viewers. If one costly expression dominates, simplify that line or its provider query rather than shortening every menu’s interval.

Check the result

Keep an interval that updates the important value within an acceptable delay without introducing repeated errors or measurable excess work. Test reopening the menu as well as leaving it open during the change.

If the test fails

Restore the old interval if performance worsens or the menu stops updating reliably. Preserve the provider and viewer-count notes for further diagnosis; repeated general reloads make the comparison harder to interpret.

The fastest visible refresh is not automatically the right choice. Tie the interval to the information players need while that particular screen remains open.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Find Minecraft logs and send useful troubleshooting details

Repair one DeluxeMenus YAML edit while preserving the working menu

A quoting or indentation mistake can prevent a menu from loading after a small edit. Preserve the failing file as evidence and use the last working copy to identify the actual difference.

Prepare the test

Work on a copy of one menu while the existing player route remains available. Use ordinary test accounts to check requirements; an administrator can bypass checks that affect your players.

Make the change

1. Find the first relevant DeluxeMenus parsing error in the console or log. Record the filename and indicated location, but inspect the preceding lines too because an unclosed quote can move the reported error.
2. Open the copied file in a text editor that shows indentation. Compare the changed section with the working version, including list markers, quotation marks and any colon inside a text value.
3. Fix one structural difference in the copied menu and use the plugin’s documented refresh route. Check whether the error disappears before changing item materials, permissions or unrelated placeholders.
4. Open the repaired menu as an ordinary player and test the changed item. Keep the prior public copy available until opening, visibility and the defined click action all pass.

Check the result

The menu should load without the original parsing error and preserve the unaffected items. Review the log after the first real opening so a runtime placeholder or action error does not hide behind

successful YAML parsing.

If the test fails

If the error remains, restore the last working menu and attach a redacted minimal section to support. Do not replace the entire DeluxeMenus folder with newly generated defaults or discard unrelated menu files.

Keep secrets out of online syntax checkers. A local editor can show indentation and quoting without publishing private command arguments or integration details.

> CloudExa game panel Files page for opening and reviewing server configuration files
> Use Files to locate your installed plugin’s own configuration. This panel example does not show the plugin-specific settings.

References and related help

Check the official DeluxeMenus documentation against your installed release before using a command or setting from another version.

- Build a safe starter menu with DeluxeMenus
- Find Minecraft logs and send useful troubleshooting details