HuskHomes tasks

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

- Resolve conflicting HuskHomes home limits across ranks
- Explain why a HuskHomes teleport has no warmup
- Disable a conflicting HuskHomes command without removing homes
- Limit a HuskHomes warp to the intended player group
- Control public HuskHomes listings without deleting private homes
- Check a HuskHomes home-slot charge before changing balances

Resolve conflicting HuskHomes home limits across ranks

A player can inherit several home allowances through donor, staff and temporary groups. Changing the number in the main configuration may appear ineffective because an explicit numeric permission takes precedence. Check the effective allowance before touching any stored home.

Before you change settings

Record the player’s normal groups, context and existing home count. Choose a test account with the same memberships and no operator status. Save the relevant permission-group definitions; the home database is not the setting being repaired.

Follow the request through the plugin

1. Inspect effective permissions for the affected account using your permission manager. Look for every granted `huskhomes.max_homes.` numeric node, including inherited and contextual entries.
2. Compare the result with the installed HuskHomes configuration. The documented default selects the largest applicable allowance; enabling `general.stack_permission_limits` changes how numeric home limits combine.
3. Choose one deliberate policy: a total allowance for each rank, or additive allowances with stacking enabled. Remove only a redundant allowance grant after confirming other groups do not depend on it.
4. Test permission evaluation in the same world and server where the complaint occurred. Use a new harmless home name only if the test account has capacity; record whether the next home is permitted or refused.

Confirm the intended result

The ordinary test account should receive the intended total, and its existing home list should remain intact. Repeat after reconnecting and after removing any temporary rank. A successful administrator test alone does not show that a member’s inherited limit is correct.

If the test fails

An unexpected larger total usually means another applicable node remains or stacking is active. A smaller total can indicate a missing context or a negative permission. Changing limits is not a reason to delete homes that already exist above the new allowance.

Restore the saved permission definitions if access becomes inconsistent. Preserve the home records and ask support to inspect effective permissions with the player, world and time of the failed check.

Sources and related help

- HuskHomes maintainer reference 1
- HuskHomes maintainer reference 2
- Set up personal homes and teleport requests with HuskHomes

Explain why a HuskHomes teleport has no warmup

An immediate teleport is not always a broken timer. HuskHomes supports both a permission-based warmup duration and a separate bypass. Staff accounts commonly differ from regular players, so compare like-for-like access before changing the delay for everyone.

Before you change settings

Choose one home in a safe location and record the configured warmup, affected player, server and permission context. Keep the same destination for each attempt so a different command or plugin cannot change the test halfway through.

Follow the request through the plugin

1. Check whether the affected player inherits `huskhomes.bypass_teleport_warmup`. A deliberate staff bypass should not be copied into a public rank just to make a test pass.
2. Inspect numeric `huskhomes.teleport_warmup.` permissions. HuskHomes documents that the highest applicable numeric value wins, even when stacking is enabled for other limits.
3. Compare a non-operator account with the affected account. Trigger the same HuskHomes home command, stand still and record the message and elapsed delay rather than estimating from the animation.
4. Adjust only the incorrect permission or documented warmup setting. Retest a member and a staff account, checking that intended staff access and member delays both match your policy.

Confirm the intended result

Your member account should wait the expected interval before arriving. The staff account should follow whichever exception you deliberately retain. Test again after reconnecting, because an old

session can make a group change harder to interpret.

If the test fails

If neither account follows the configured duration, confirm that HuskHomes owns the command. Another homes plugin can register the same label. A non-positive configured delay also needs checking; permission changes cannot prove which command handler actually ran.

Restore the earlier warmup and permission entries if the change affects unrelated players. Keep the timing notes and command ownership evidence; do not grant operator status as a workaround for a teleport delay.

Sources and related help

- HuskHomes maintainer reference 1
- HuskHomes maintainer reference 2
- Set up personal homes and teleport requests with HuskHomes

Disable a conflicting HuskHomes command without removing homes

Two teleport plugins can compete for labels such as home, spawn or random teleport. You can leave HuskHomes installed and disable a selected registration instead of removing its data or repeatedly changing plugin load order.

Before you change settings

Write down the exact command and aliases players use, and decide which plugin should own that action. Save HuskHomes configuration and the other plugin’s command settings. Keep a list of existing destinations for a before-and-after check.

Follow the request through the plugin

1. Confirm ownership using the command-conflict documentation and each plugin’s help. Do not assume the visible response identifies the handler when both plugins use similar messages.
2. Open the active HuskHomes configuration. Add only the unwanted slash-prefixed command to its documented `disabled_commands` list, preserving the YAML list structure and existing entries.
3. Use a planned full server restart for a registration change. A configuration reload does not provide the same assurance that the command map has been rebuilt cleanly.
4. Test the chosen public label and documented aliases as a member. Then test another HuskHomes action that should remain available and inspect its destination list.

Confirm the intended result

The competing plugin should consistently handle the selected label, while unaffected HuskHomes commands still work. Existing saved homes should remain present because disabling command registration does not require removing their records. Update player instructions if the command now has different arguments.

If the test fails

An alias can still resolve differently from its main command. Check the disabled list and any server command aliases before making further changes. If the new handler cannot find an old home, remember that separate plugins do not automatically share destination databases.

Remove the single new disabled entry and restart during a maintenance window to restore previous registration. Keep all destination data; do not import, clear or merge home databases merely to resolve a command-name collision.

Sources and related help

- HuskHomes maintainer reference 1
- HuskHomes maintainer reference 2
- Set up personal homes and teleport requests with HuskHomes

Limit a HuskHomes warp to the intended player group

A warp can exist correctly while remaining unavailable to the group that needs it. Restricted warps add a destination-specific permission check to the ordinary warp command. This is useful for staff rooms or rank areas, provided the test includes a player who should be refused.

Before you change settings

Choose a harmless test destination and two non-operator accounts: one allowed and one denied. Record the warp’s exact name and save permission-group settings. Confirm that permission-restricted warps are supported and enabled as documented for your installed release.

Follow the request through the plugin

1. Check the allowed account has the required command permission for the action being tested. Listing, travelling, editing and deleting are separate actions; granting travel should not silently grant management.
2. Grant the narrow destination permission `huskhomes.warp.` followed by the exact warp name to the intended group. Inspect inherited permissions for unintended broad access before testing.
3. Use the allowed account to list warps and travel to the destination. Run the corresponding list and travel attempts with the denied account, recording the response each receives.
4. Repeat from another relevant world or server if your permissions use contexts. Reconnect after group changes and confirm the public player instructions contain only commands that the intended group can actually use.

Confirm the intended result

The permitted member should list and use the destination, while the denied member should remain excluded under the configured restriction. Check another public warp too, so the change has not accidentally removed basic travel for ordinary players.

If the test fails

If both accounts succeed, inspect inherited destination access and whether restrictions are active. If neither succeeds, check command access before renaming the warp. Creating another warp with the same destination does not repair a permission mismatch.

Restore the saved group entries and restriction setting if access widens unexpectedly. Retain the original warp and coordinates. Ask support to inspect the effective permissions rather than publishing a restricted location or granting broad administrative access.

Sources and related help

- HuskHomes maintainer reference 1
- HuskHomes maintainer reference 2
- Set up personal homes and teleport requests with HuskHomes

Control public HuskHomes listings without deleting private homes

A total home allowance and a public listing allowance solve different problems. Increasing one does not necessarily increase the other. Check publication permissions and limits before treating an inability to publish a destination as a missing or damaged home.

Before you change settings

Ask the player which existing home they want to publish and confirm that its coordinates are suitable for public visitors. Record current privacy states and numeric permissions. Use a test home away from private builds when checking the visitor experience.

Follow the request through the plugin

1. Inspect `huskhomes.max_public_homes.` numeric grants separately from total home allowances. Check inherited group limits and the configured stacking policy rather than changing every allowance together.
2. Confirm that the member has the documented home-edit and privacy permissions. The ability to create a home does not prove that every edit subcommand is available.
3. In game, use `/edithome` with the home name and the documented privacy action to change only the chosen test home. List public homes from another ordinary account.
4. Visit the new public listing with that second account and check the destination. Then return the test home to private and confirm it no longer appears as an available public destination.

Confirm the intended result

The public list should change for the selected home while other private homes stay private and remain in the owner’s personal list. Record how the public allowance is counted after reversing the test so the player understands which entries occupy a public slot.

If the test fails

If publication fails despite spare total capacity, recheck the public allowance and privacy permission. If the entry appears to the owner but not a visitor, distinguish the owner’s personal list from the shared public list before changing data.

Restore the test home’s original privacy and the saved numeric grants. Avoid deleting and recreating homes to free a listing slot; that can lose descriptions, ownership context and the original location while leaving the underlying policy unresolved.

Sources and related help

- HuskHomes maintainer reference 1
- HuskHomes maintainer reference 2
- Set up personal homes and teleport requests with HuskHomes

Check a HuskHomes home-slot charge before changing balances

An additional home slot can involve a maximum allowance, a free allowance and previously purchased capacity. A player being charged unexpectedly does not justify editing their balance or removing a saved destination. Follow the slot decision first.

Before you change settings

Record the installed build, existing homes, current balance and the exact attempted action. Save the economy configuration and relevant permission entries. On Paper, verify Vault and the intended economy provider loaded successfully before diagnosing a home purchase.

Follow the request through the plugin

1. Inspect the member’s numeric free-home and maximum-home permissions. These may override configuration defaults, and inherited memberships can make two apparently identical ranks behave differently.
2. Use the documented `/huskhomes homeslots` view action for that player to inspect purchased capacity. View the result before considering any administrative adjustment; do not repeatedly run a purchase as a diagnostic.
3. Check the additional-slot cost in the generated configuration for the installed build. The maintainer pages show release-dependent naming, so retain the file’s real structure instead of pasting a guessed YAML key.
4. Check whether the test account has `huskhomes.bypass_economy_checks`. Run one controlled purchase only with the owner’s agreement, recording balance and slot count immediately before and after.

Confirm the intended result

A successful purchase should produce the intended slot increase and a single matching debit. An account with a deliberate bypass should follow its documented exception. Confirm that a later home creation uses the available capacity rather than charging the same purchase again.

If the test fails

If the balance changes without usable capacity, preserve timestamps and economy-provider logs for support. Separate a failed provider hook from a maximum-home limit. Do not compensate speculatively while an automatic or queued transaction may still complete.

Restore only the changed allowance or price after the test. If an adjustment is genuinely required, reconcile the recorded transaction and slot count first; never restore an old entire economy database over unrelated players’ later activity.

Sources and related help

- HuskHomes maintainer reference 1
- HuskHomes maintainer reference 2
- Set up personal homes and teleport requests with HuskHomes