Player Search & Robbery

Player Search & Robbery

Server-authoritative player-to-player viewing and one-way transfer.

Applies to: `Nord Inventory v2.0.0`

Player Search & Robbery

Nord Inventory can expose **Search player** and **Rob player** interactions when the target player is surrendering with their hands up.

Security model

The target option is only a client-side convenience. The server revalidates:

- target server ID;
- distance between both players;
- the target's replicated hands-up state;
- the mode (`search` or `rob`);
- every inventory transfer while the session is open.

The session closes automatically if the target lowers their hands, disconnects, or moves outside the configured distance.

Search

Search is read-only.

The searching player can inspect the target inventory, including items stored in Fast Slots and Special Slots, but cannot move items from/to the target inventory.

Rob

Robbery is one-way:

```
victim inventory  ->  robber inventory
```

The robber can never use the session to put/swap their own items into the victim inventory.

Pinned Fast Slot items are not protected from a valid robbery. If stolen, the victim's pin marker is removed before the item enters the robber inventory.

Configuration

Edit:

```
config/player_interactions.lua
```

Main options:

```
Config.PlayerRobbery = {
    Enabled = true,
    Distance = 2.5,
    SearchEnabled = true,
    RobEnabled = true,
    RequireHandsUp = true
}
```

Hands-up detection

Nord automatically detects common hands-up animation dictionaries and publishes:

```
nord_inventory:handsup
```

as a replicated player state.

If your hands-up resource uses a custom animation, add it to `Animations`.

If it already publishes a player state, add the key to `ExternalStateKeys`.

An external client resource can also explicitly tell Nord:

```
exports['nord_inventory']:SetHandsUpState(true)
```

Return to automatic detection:

```
exports['nord_inventory']:SetHandsUpState('auto')
```

Check the local state:

```
local handsUp = exports['nord_inventory']:IsHandsUp()
```

Target providers

When the inventory interaction mode uses Target, Nord registers global-player options for ox_target / qb-target / compatible providers.

When Target is not active, the TextUI fallback uses:

```
E -> Search
G -> Rob
```

The controls and labels are configurable.

Server exports

Trusted server resources can request the same validated flow:

```
exports['nord_inventory']:OpenPlayerSearch(source, target)
exports['nord_inventory']:OpenPlayerRobbery(source, target)
exports['nord_inventory']:IsPlayerSurrendered(target)
```

Reliable interaction mode

Player Search/Rob now has its own interaction selector and does not depend on `Config.Interaction.Mode`:

```
Config.PlayerRobbery.Interaction = 'auto'
```

- `auto`: use ox_target/qb-target when available, otherwise TextUI;
- `target`: target only;
- `textui`: TextUI only.

This prevents a world-interaction setting from silently disabling player robbery.

Built-in hands up

The system now works without a separate surrender resource:

```
X -> toggle hands up
/handsup -> toggle hands up
```

Configure or disable it in `Config.PlayerRobbery.BuiltInHandsUp`.

Remote Search/Rob options only appear after the target's replicated hands-up state is visible. This ensures the server and the robber agree on the surrender state before the interaction can be selected.

Server-synchronized surrender state

Hands-up detection is performed on the surrendering player's own client and sent to the server. The server then broadcasts a compact surrender cache to all clients.

Target/TextUI checks use that server-synchronized cache. This avoids differences between FiveM artifacts in how remote Player state bags and ped Entity state bags are exposed.

A joining/restarting client requests a full snapshot automatically.

Revision #1
Created 2026-09-30 01:04:36 UTC by Riguita
Updated 2026-09-30 01:04:36 UTC by Riguita