8 · Database, Security & Troubleshooting

Operate the resource safely in production and resolve the most common installation issues.

- Database Schema
- Security Model
- Production Checklist
- Common Issues
- Bridge Troubleshooting
- FAQ

Database Schema

Auto SQL creates the resource schema automatically when persistence is enabled.

Core tables:

`nord_vehiclekeys_keys`

Stores persisted key grants:

- plate
- holder identifier
- key type
- expiry timestamp
- created/updated timestamps

`nord_vehiclekeys_recovery_npcs`

Stores persistent recovery NPC placement and appearance:

- label
- ped model
- scenario
- X/Y/Z
- heading
- creator identifier

`nord_vehiclekeys_schema`

Tracks internal database schema revisions. This is a database schema number, not the resource release number.

Manual SQL is included at:

```
sql/install.sql
```

Security Model

Nord Vehicle Keys is designed around server authority.

Important protections

- OneSync entity checks are required by default.
- Lock state is synchronized through entity state bags.
- Key grants/removals are validated server-side.
- Physical key metadata is re-read from the authoritative inventory slot.
- Session/stolen keys are bound to the player connection/session token.
- Carjacking actions use server-side action reservation and validation.
- Ownership lookup is server-side.
- Recovery keys are only issued after ownership validation.
- Admin actions use ACE permission checks.
- Sensitive compatibility events are disabled by default.
- Callback/action rate limits reduce spam and replay abuse.

Trusted export rule

Exports such as `GiveKeys` should only be called by trusted server resources after that resource has validated its own business logic.

Production Checklist

Before production deployment:

- [ ] OneSync enabled.
- [ ] Database backups available.
- [ ] `oxmysql` starts before Nord Vehicle Keys.
- [ ] Framework bridge detected correctly.
- [ ] Inventory bridge selected correctly.
- [ ] `vehicle_key` definition is non-stackable when using Nord Inventory.
- [ ] Recovery ACE configured only for staff who need it.
- [ ] Ownership table/columns match your framework schema.
- [ ] Custom bridge code returns valid values and does not trust arbitrary client payloads.
- [ ] Test owner key issue/removal.
- [ ] Test player key sharing and revocation.
- [ ] Test stolen key removal after exiting the vehicle.
- [ ] Test recovery NPC key reissue.
- [ ] Test ghost placement cancel/confirm.
- [ ] Test lock state with multiple players nearby.

Common Issues

Physical key exists but does not open the remote

Check that:

- `nord_inventory` starts before `nord_vehiclekeys`.
- The `vehicle_key` item is non-stackable.
- The slot metadata contains a normalized plate.
- The key is not expired.
- The physical-key item use handler is registered.

Player keeps getting a key back after deleting it

Strict physical-key mode should use:

```
RequirePhysicalItemForAccess = true
OwnerAccessWithoutPhysicalItem = false
StoredAccessWithoutPhysicalItem = false
AllowClientOwnerClaim = false
```

A missing key should then only return through recovery/admin/trusted server exports.

Recovery NPC is underground

Use the visual placement editor and save the final position. The saved Z is authoritative and the final NPC should not run a second ground snap.

Recovery NPC is stuck in a broken scenario pose

Use a valid GTA scenario name. The resource applies the scenario without a destructive intro sequence and freezes the NPC after the final placement is applied.

Fullscreen dark NUI overlay

The key remote NUI should be transparent and hidden while closed. If you replaced the web files, ensure `html` and `body` do not force a dark background or `color-scheme: dark`.

"The action is no longer valid" during carjacking

Make sure all resource files are from the same build and that OneSync is enabled. The carjack completion validates the reserved vehicle/action rather than relying on the NPC remaining in the same network seat state.

Bridge Troubleshooting

Framework not detected

Force the bridge in `config/bridges.lua`:

```
Config.Bridges.Framework = 'qb' -- or qbx / esx / standalone / custom
```

Inventory not detected

Force the bridge:

```
Config.Bridges.Inventory = 'nord' -- or ox / qb / esx / custom
```

If `Config.Inventory.Strict = true`, item-requiring actions are denied when no bridge can validate the item.

Custom notification not showing

Set:

```
Config.Bridges.Notify = 'custom'
```

and make `NVKNotify.Custom(...)` return `true` after showing the notification.

Custom Text UI not hiding

Both custom hooks must be implemented:

```
NVKTextUI.CustomShow(...)
NVKTextUI.CustomHide(...)
```

Return `true` when the action was handled.

FAQ

Does owning a vehicle automatically recreate a deleted key?

No, not with the default strict physical-key settings. The player must recover or be explicitly issued a new key.

Can I use the resource without Nord Inventory?

Yes for generic access/item checks through supported bridges. The complete physical `vehicle_key` metadata workflow is native to Nord Inventory; a custom inventory can implement the provided physical-key hooks.

Can a player recover a key for a vehicle they do not own?

No. Recovery validates ownership server-side before issuing the key.

Do stolen keys stay forever?

No. They are designed to expire when the player leaves the stolen vehicle and also have a safety timeout.

Can a garage or dealership give keys?

Yes. Use the trusted server export:

```
exports.nord_vehiclekeys:GiveKeys(source, plate)
```

Can I replace notifications or Text UI without editing the core?

Yes. Select `custom` in `config/bridges.lua` and edit only the matching bridge file.

Is the recovery NPC placement persistent?

Yes. NPC definitions and coordinates are stored in the database.