

Manuelle Installation im Docker Container

Diese Anleitung beschreibt, wie Sie den LibreBooking Node manuell im Docker Container installieren, wenn die Dateien bereits kopiert wurden, aber der Node nicht in n8n erscheint.

Situation

Sie haben die Dateien in den Container kopiert (z.B. nach `/opt/n8n/custom-nodes` oder `/home/node/.n8n/custom/n8n-nodes-librebooking`), aber:

- Der LibreBooking Node erscheint nicht in der n8n Node-Suche
- Die TypeScript-Dateien wurden nicht kompiliert
- Das `dist/` Verzeichnis fehlt oder ist leer

Diagnose

Schnell-Check mit einem Befehl

```
# Prüft Dateien und Status im Container
docker exec n8n sh -c "ls -la /home/node/.n8n/custom/*/dist/ 2>/dev/null || echo
'dist/ nicht gefunden'"
```

Ausführlicher Check mit Skript

```
# Check-Skript in Container kopieren und ausführen
docker cp check-installation.sh n8n:/tmp/
docker exec n8n sh /tmp/check-installation.sh
```

Manuell im Container prüfen

```
# In den Container einloggen
docker exec -it n8n sh

# Prüfen, was vorhanden ist
ls -la /home/node/.n8n/custom/
ls -la /home/node/.n8n/custom/n8n-nodes-librebooking/

# Umgebungsvariablen prüfen
env | grep N8N

# Container verlassen
exit
```

Lösung 1: Automatisch mit Skript (empfohlen)

Das einfachste ist das All-in-One-Skript:

```
# Auf dem Host ausführen
./fix-node-installation.sh
```

Oder mit spezifischem Container-Namen:

```
./fix-node-installation.sh -c mein-n8n-container
```

Alternative: Nur das Installations-Skript

```
# Skript in Container kopieren
docker cp install-in-container.sh n8n:/tmp/

# Skript im Container ausführen
docker exec -it n8n sh /tmp/install-in-container.sh

# Container neustarten
docker restart n8n
```

Lösung 2: Manuelle Schritte im Container

Schritt 1: In den Container einloggen

```
docker exec -it n8n sh
```

Schritt 2: Zum Custom-Node-Verzeichnis wechseln

```
# Standard-Pfad (offizielles n8n Image)
cd /home/node/.n8n/custom/n8n-nodes-librebooking

# Oder falls an anderem Ort:
cd /opt/n8n/custom-nodes
# oder
cd /data/custom-nodes
```

Schritt 3: Prüfen ob Dateien vorhanden sind

```
ls -la
# Sollte zeigen: package.json, tsconfig.json, nodes/, credentials/
```

Schritt 4: Dependencies installieren

```
npm install
```

Erwartete Ausgabe:

```
added 50 packages in 10s
```

Schritt 5: Node komplizieren

```
npm run build
```

Erwartete Ausgabe:

```
> n8n-nodes-librebooking@1.0.0 build
> tsc && npm run copy:icons
```

Schritt 6: Prüfen ob Build erfolgreich war

```
ls -la dist/
ls -la dist/nodes/LibreBooking/
```

Sollte zeigen:

- dist/nodes/LibreBooking/LibreBooking.node.js
- dist/nodes/LibreBookingTrigger/LibreBookingTrigger.node.js
- dist/credentials/LibreBookingApi.credentials.js

Schritt 7: Container verlassen und neustarten

```
# Container verlassen
exit

# Container neustarten (auf dem Host)
docker restart n8n
```

Lösung 3: Direkt mit docker exec (Ein-Befehl-Lösung)

Wenn Sie schnell zum Ziel kommen wollen:

```
# Alles in einem Befehl
docker exec n8n sh -c
"cd /home/node/.n8n/custom/n8n-nodes-librebooking && npm install && npm run build"

# Container neustarten
docker restart n8n
```

Für anderen Pfad:

```
docker exec n8n sh -c "cd /opt/n8n/custom-nodes && npm install && npm run build"
docker restart n8n
```

Verifizierung

1. Logs prüfen

```
# Nach LibreBooking in den Logs suchen
docker logs n8n 2>&1 | grep -i "librebooking\|custom\|node"
```

2. In n8n prüfen

1. Öffnen Sie n8n im Browser (z.B. `http://localhost:5678`)
2. Erstellen Sie einen neuen Workflow oder öffnen Sie einen bestehenden
3. Klicken Sie auf `+` um einen neuen Node hinzuzufügen
4. Suchen Sie nach "LibreBooking"
5. Es sollten zwei Nodes erscheinen:
 - **LibreBooking** - Hauptnode für alle Operationen
 - **LibreBooking Trigger** - Trigger für Reservierungen

3. Node-Dateien im Container prüfen

```
# Prüfe ob .node.js Dateien existieren
docker exec n8n find /home/node/.n8n/custom -name "*.*.node.js" 2>/dev/null
```

Troubleshooting

Problem: npm nicht gefunden

Symptom:

```
sh: npm: not found
```

Lösung:

Das verwendete Docker-Image enthält kein npm. Verwenden Sie ein Image mit Node.js:

```
# Prüfen Sie das Image
docker inspect n8n --format='{{.Config.Image}}'

# Das offizielle n8n Image (n8nio/n8n) enthält npm
# Falls Sie ein minimales Image verwenden, wechseln Sie zu n8nio/n8n
```

Problem: Permission denied

Symptom:

```
EACCES: permission denied
npm ERR! could not create a lockfile
```

Lösung:

n8n läuft als User `node` (UID 1000). Setzen Sie die Berechtigungen:

```
# Auf dem Host (vor dem Kopieren)
sudo chown -R 1000:1000 custom-nodes/

# Oder im Container als root
docker exec -u root n8n chown -R node:node /home/node/.n8n/custom/
```

Problem: Build schlägt fehl

Symptom:

```
error TS2307: Cannot find module 'n8n-workflow'
```

Lösung:

```
# Im Container
cd /home/node/.n8n/custom/n8n-nodes-librebooking
rm -rf node_modules package-lock.json
npm install
npm run build
```

Problem: Node erscheint nach Neustart immer noch nicht

Mögliche Ursachen:

1. **Falscher Pfad:** N8N_CUSTOM_EXTENSIONS zeigt nicht auf das richtige Verzeichnis

```
bash
docker exec n8n env | grep N8N_CUSTOM
# Sollte zeigen: N8N_CUSTOM_EXTENSIONS=/home/node/.n8n/custom
```

2. **package.json n8n-Sektion fehlt:** Die n8n Konfiguration in package.json muss korrekt sein

```
bash
docker exec n8n cat /home/node/.n8n/custom/n8n-nodes-librebooking/package.json | grep -A10 '"n8n"'
```

3. **Container-Cache:** Container komplett neu erstellen

```
bash
docker compose down
docker compose up -d
```

4. **n8n Version zu alt:** Der Node benötigt n8n >= 1.0.0

```
bash
docker exec n8n n8n --version
```

Problem: Fehler bei npm install (Netzwerk)

Symptom:

```
npm ERR! network request failed
```

Lösung:

```
# DNS prüfen
docker exec n8n cat /etc/resolv.conf

# npm Registry prüfen
docker exec n8n npm config get registry

# Ggf. Registry setzen
docker exec n8n npm config set registry https://registry.npmjs.org/
```

Schnellreferenz

Aktion	Befehl
Status prüfen	docker exec n8n sh /tmp/check-installation.sh
Installieren	docker exec n8n sh -c "cd /home/node/.n8n/custom/n8n-nodes-librebooking && npm install && npm run build"
Neustarten	docker restart n8n
Logs prüfen	docker logs n8n 2>&1 \ grep -i libre
Im Container	docker exec -it n8n sh

Weiterführende Dokumentation

- [DOCKER-INTEGRATION.md](#) (DOCKER-INTEGRATION.md) - Vollständige Docker-Anleitung
- [TROUBLESHOOTING.md](#) (TROUBLESHOOTING.md) - Alle Troubleshooting-Themen
- [SCHNELLSTART-DOCKER.md](#) (SCHNELLSTART-DOCKER.md) - Docker-Schnellstart

Letzte Aktualisierung: Januar 2026