

Troubleshooting

Inhaltsverzeichnis

1. [tsc not found](#)
 2. [Read-only Volume](#)
 3. [npm audit Vulnerabilities](#)
 4. [Node nicht sichtbar](#)
 5. [Authentifizierung fehlgeschlagen](#)
-

tsc not found

Symptom:

```
sh: 1: tsc: not found
npm error code 127
```

Ursache: TypeScript ist nicht installiert (im Container oft nicht verfügbar).

Lösung: Auf dem Host bauen!

```
# Auf dem Host (nicht im Container)
npm install
npm run build

# Dann in Container kopieren
docker cp dist n8n:/home/node/.n8n/custom/n8n-nodes-librebooking/
docker cp package.json n8n:/home/node/.n8n/custom/n8n-nodes-librebooking/
docker cp node_modules n8n:/home/node/.n8n/custom/n8n-nodes-librebooking/
docker restart n8n
```

Oder mit Skript:

```
./quick-install.sh n8n
```

Read-only Volume

Symptom:

```
EROFS: read-only file system
npm error EACCES: permission denied
```

Ursache: Volume ist mit `:ro` gemountet.

Lösung 1: Volume ohne `:ro` mounten

```
# docker-compose.yml
volumes:
  - ./custom-nodes:/home/node/.n8n/custom # Ohne :ro
```

Lösung 2: Auf dem Host bauen und kopieren

```
npm install && npm run build
docker cp dist n8n:/home/node/.n8n/custom/n8n-nodes-librebooking/
```

npm audit Vulnerabilities

Symptom:

```
found 2 vulnerabilities (1 moderate, 1 critical)
```

Erklärung: Diese kommen von `n8n-workflow` und sind für dieses Projekt nicht kritisch.**Lösung:** Ignorieren oder `.npmrc` verwenden:

```
# .npmrc bereits konfiguriert
audit=false
```

Siehe [SECURITY.md](#) (SECURITY.md) für Details.

Node nicht sichtbar

Lösung:

1. Container neustarten:

```
bash
```

```
docker restart n8n
```

2. Dateien prüfen:

```
bash
```

```
docker exec n8n ls -la /home/node/.n8n/custom/n8n-nodes-librebooking/
```

3. Environment prüfen:

```
```yaml
```

```
environment:
```

- `N8N_CUSTOM_EXTENSIONS=/home/node/.n8n/custom`
- ```
```
```

4. Logs prüfen:

```
bash
```

```
docker logs n8n | grep -i librebooking
```

---

## Authentifizierung fehlgeschlagen

---

**Symptom:** “401 Unauthorized” oder “Invalid credentials”

**Lösung:**

1. URL prüfen (ohne `/Web/` am Ende)
2. Benutzer muss Admin sein
3. API muss in LibreBooking aktiviert sein

Test:

```
curl -X POST https://librebooking.example.com/Web/Services/Authentication/Authentic-
ate \
-H 'Content-Type: application/json' \
-d '{"username": "admin", "password": "pass"}'
```

---

## Weitere Hilfe

---

- [README.md](#) (README.md) - Übersicht
- [INSTALLATION.md](#) (INSTALLATION.md) - Installationsanleitung
- [DOCKER-INTEGRATION.md](#) (DOCKER-INTEGRATION.md) - Docker Details