🔒 Security Fixes Report

Datum: 22. Januar 2026
PR: #16 (https://github.com/MathiasHeinke/leanaf/pull/16)
Branch: `security/critical-fixes`

Zusammenfassung

Diese Migration behebt **kritische Sicherheitslücken** im LeanAF Supabase-Backend, die bei einem Security-Audit identifiziert wurden.

🚨 Identifizierte Probleme

1. Funktionen ohne `search_path` (KRITISCH)

Risiko: SQL-Injection durch search_path Manipulation
Gefundene Funktionen ohne search_path:
- `update_updated_at_column()` - Trigger für alle Tabellen
- `handle_new_user()` - Auth-Trigger (HOCHKRITISCH - wird bei jedem neuen User ausgeführt)
- `is_super_admin()` - Admin-Prüfung
- `is_enterprise_or_super_admin()` - Admin-Prüfung
- `is_super_admin_by_email()` - Admin-Prüfung
- `check_ai_usage_limit()` - Rate Limiting
- `has_admin_access()` - Admin-Prüfung

Status: ✅ GEFIXT

2. `subscribers` Tabelle mit USING(true) (KRITISCH)

Risiko: Jeder konnte fremde Subscriptions ändern/einsehen
Vorher:

```
CREATE POLICY "update_own_subscription" ON public.subscribers
FOR UPDATE USING (true);  -- GEFÄHRLICH!

CREATE POLICY "insert_subscription" ON public.subscribers
FOR INSERT WITH CHECK (true);  -- GEFÄHRLICH!
```

Nachher:

```
CREATE POLICY "subscribers_select_own" ON public.subscribers
FOR SELECT TO authenticated
USING (user_id = auth.uid() OR email = auth.email());

CREATE POLICY "subscribers_update_own" ON public.subscribers
FOR UPDATE TO authenticated
USING (user_id = auth.uid() OR email = auth.email());
```

Status: ✅ GEFIXT

3. Fehlende RLS auf sensitiven Tabellen

Risiko: Direkter Datenzugriff ohne RLS-Prüfung möglich
Betroffene Tabellen:
- `coach_memory` - AI Konversationskontext
- `ares_traces` - AI Interaktionslogs
- `ai_usage_logs` - Nutzungsstatistiken

Status: ✅ GEFIXT (RLS enabled + proper policies)

4. Anon-Zugriff auf sensitive Tabellen

Risiko: Nicht authentifizierte User könnten auf Daten zugreifen
Betroffene Tabellen:
- `profiles` - Enthält PII (Email, Gewicht, Alter)
- `subscribers` - Abo-Informationen
- `admin_users` - Admin-Liste

Status: ✅ GEFIXT (REVOKE ALL FROM anon)

📋 Änderungen im Detail

Geänderte/Erstellte Datei

```
supabase/migrations/20260122160000_security_critical_fixes.sql
```

Teil 1: Funktionen mit search_path

Alle kritischen Funktionen wurden aktualisiert mit:

```
SECURITY DEFINER
SET search_path = public
```

Teil 2: RLS Aktivierung

```
ALTER TABLE IF EXISTS public.profiles ENABLE ROW LEVEL SECURITY;
ALTER TABLE IF EXISTS public.workouts ENABLE ROW LEVEL SECURITY;
ALTER TABLE IF EXISTS public.meals ENABLE ROW LEVEL SECURITY;
-- ... weitere Tabellen
```

Teil 3: Sichere Policies für `subscribers`

- User können nur eigene Daten sehen/ändern
- Service Role behält vollen Zugriff (für Stripe Webhooks)
- Admins können alle Subscriptions verwalten

Teil 4: Sichere Policies für `coach_memory`

- User sehen nur eigenen Kontext
- Service Role für Edge Functions

- Admins für Debugging

Teil 5: Sichere Policies für `ares_traces`

- User sehen nur eigene Traces
- Service Role für Logging
- Admins für Debugging

Teil 6: Permissions

```
REVOKE ALL ON public.profiles FROM anon;
REVOKE ALL ON public.subscribers FROM anon;
REVOKE ALL ON public.coach_memory FROM anon;
REVOKE ALL ON public.ares_traces FROM anon;
REVOKE ALL ON public.ai_usage_logs FROM anon;
REVOKE ALL ON public.admin_users FROM anon;
```

⚠️ Nicht behobene Issues (bewusst beibehalten)

USING(true) für öffentliche Daten

Folgende Tabellen behalten `USING(true)` für SELECT, da sie öffentliche/Lookup-Daten enthalten:
- `exercises` - Übungskatalog
- `food_database` - Lebensmitteldaten
- `coach_knowledge_base` - RAG Wissensbasis
- `knowledge_base_embeddings` - RAG Embeddings
- `feature_requests` - Community Feature-Voting

Diese sind **bewusst öffentlich** für die App-Funktionalität.

📊 Verbleibende Funktionen ohne search_path

Es gibt noch ~170+ Funktionen ohne expliziten search_path. Diese wurden nicht alle gefixt, da:
1. Viele sind Supabase-interne Funktionen
2. Trigger-Funktionen mit geringem Risiko
3. Nicht alle in auth-kritischen Pfaden

Empfehlung: Bei zukünftigen Funktionen immer `SET search_path = public` verwenden.

🚀 Deployment-Anleitung

1. Migration anwenden

```
cd leanaf
supabase db push
```

2. Verifizieren

```
-- Prüfe RLS Status
SELECT schemaname, tablename, rowsecurity
FROM pg_tables
WHERE schemaname = 'public'
AND tablename IN ('profiles', 'subscribers', 'coach_memory', 'ares_traces');

-- Sollte ausgeben:
-- public | profiles     | true
-- public | subscribers  | true
-- public | coach_memory | true
-- public | ares_traces  | true
```

3. Edge Functions neu deployen

```
supabase functions deploy coach-orchestrator-enhanced
```

✅ Checkliste für Review

- [ ] Migration auf Staging testen
- [ ] Stripe Webhooks funktionieren noch
- [ ] Coach Chat funktioniert noch
- [ ] Neue User können sich registrieren
- [ ] Admin-Panel funktioniert
- [ ] Keine fremden Daten sichtbar

📝 Fazit

Diese Migration behebt die **kritischsten** Sicherheitsprobleme:

Problem	Schweregrad	Status
Funktionen ohne search_path	HOCH	✅ Gefixt
subscribers USING(true)	KRITISCH	✅ Gefixt
coach_memory ungesichert	HOCH	✅ Gefixt
ares_traces ungesichert	MITTEL	✅ Gefixt
Anon-Zugriff	MITTEL	✅ Gefixt

Nächste Schritte:
1. PR reviewen und auf Staging testen
2. Nach erfolgreichen Tests auf Production deployen
3. Security-Audit für verbleibende Funktionen planen