

# Error de Conexión a Base de Datos en Vercel

## Resumen del Problema

La aplicación **Rental Management** desplegada en Vercel no puede conectarse a la base de datos PostgreSQL alojada en Abacus.AI.

## Error Observado

```
Invalid `prisma.tenant.findUnique()` invocation:  
Can't reach database server at `db-43c1c84ad.db003.hosteddb.reai.io:5432`  
Please make sure your database server is running at  
`db-43c1c84ad.db003.hosteddb.reai.io:5432`.
```

## Análisis Realizado

### 1. Verificación de Conexión Local

La conexión desde el entorno local funciona perfectamente:

 Conexión exitosa. Tenants encontrados: 4

### 2. Verificación de Variables de Entorno

Se actualizó correctamente la variable `DATABASE_URL` en Vercel con:

```
postgresql://role_43c1c84ad:***@db-43c1c84ad.db003.hosteddb.reai.io:5432/43c1c84ad?  
connect_timeout=15&pool_timeout=20&connection_limit=10
```

### 3. Optimización de Prisma Client

Se mejoró el archivo `lib/prisma.ts` con:

- Singleton pattern correcto
- Configuración optimizada para serverless
- Gestión apropiada de conexiones

### 4. Test de Conectividad de Red

```
Connection to db-43c1c84ad.db003.hosteddb.reai.io (172.21.254.220) 5432 port succeeded
```

 **PROBLEMA IDENTIFICADO:** La base de datos resuelve a una **IP privada** ( 172.21.254.220 )



## Causa Raíz

La base de datos de Abacus.AI está en una **red interna/privada** que:

- Es accesible desde entornos de Abacus.AI (como este servidor)
- **NO es accesible desde servicios externos** como Vercel

Las funciones serverless de Vercel se ejecutan en infraestructura de AWS/Google Cloud con IPs públicas que no tienen acceso a la red privada de Abacus.AI.



## Soluciones Posibles

### Opción 1: Migrar a Base de Datos Pública (RECOMENDADO)

Usar un proveedor de PostgreSQL diseñado para aplicaciones serverless:

#### A) Neon.tech (Recomendado)

##### Ventajas:

- Optimizado para serverless (connection pooling nativo)
- Tier gratuito generoso (3GB storage, 500MB RAM)
- Branching de bases de datos para desarrollo
- Escala a 0 automáticamente
- Compatible con Prisma

**URL:** <https://neon.tech>

**Tiempo:** 10 minutos

#### B) Supabase

##### Ventajas:

- PostgreSQL completo + APIs REST automáticas
- Tier gratuito (500MB database, 50k usuarios)
- Connection pooler incluido
- Dashboard amigable

**URL:** <https://supabase.com>

**Tiempo:** 15 minutos

#### C) Railway

##### Ventajas:

- PostgreSQL estándar
- Deploy fácil
- \$5 de créditos gratis/mes

**URL:** <https://railway.app>

**Tiempo:** 10 minutos

## Proceso de Migración

```
# 1. Exportar datos actuales de Abacus.AI
cd /home/ubuntu/rental_management
npx prisma db pull
pg_dump $DATABASE_URL > backup.sql

# 2. Crear nueva base de datos en Neon/Supabase/Railway

# 3. Actualizar DATABASE_URL con la nueva conexión

# 4. Importar datos
psql $NEW_DATABASE_URL < backup.sql

# 5. Ejecutar migraciones
npx prisma migrate deploy

# 6. Actualizar variable en Vercel
npx vercel env rm DATABASE_URL production
echo "nueva_url" | npx vercel env add DATABASE_URL production

# 7. Redeploy
npx vercel --prod
```

---

## Opción 2: Prisma Data Proxy (Intermedio)

Usar Prisma Accelerate como proxy de conexiones:

**Ventajas:**

- Connection pooling profesional
- Caching de queries
- Funciona con BD privadas

**Desventajas:**

-  Servicio de pago (\$25/mes mínimo)
-  Configuración adicional

**URL:** <https://www.prisma.io/data-platform/accelerate>

---

## Opción 3: Cloudflare Tunnel (Avanzado)

Crear un túnel público hacia la base de datos privada:

**Ventajas:**

- Mantiene la BD actual
- Seguro (autenticado)

**Desventajas:**

-  Requiere servidor siempre encendido
-  Configuración compleja
-  Punto único de fallo

**Complejidad:** Alta**Mantenimiento:** Continuo

## Opción 4: Servidor Proxy Intermedio (Complejo)

Desplegar un servidor Node.js en la misma red de Abacus.AI:

**Arquitectura:**

Vercel → API Gateway (público) → Proxy Server (Abacus.AI) → Database

**Ventajas:**

- Control total

**Desventajas:**

-  Infraestructura adicional
-  Latencia añadida
-  Costo de mantenimiento

**Complejidad:** Muy Alta

## Opción 5: Contactar Soporte de Abacus.AI

Preguntar si pueden:

- Asignar una IP pública a la base de datos
- Permitir acceso desde rangos de IP de Vercel
- Proporcionar connection pooler público

**Probabilidad de éxito:** Baja (las bases de datos internas generalmente están diseñadas para uso interno)

## 🎯 Recomendación Final

### ⭐ Solución Óptima: Migrar a Neon.tech

**Razones:**

1. **Diseñado para serverless** - Zero conexiones cuando no se usa
2. **Connection pooling nativo** - No más timeouts
3. **Tier gratuito suficiente** - 3GB storage, ideal para empezar
4. **Prisma-friendly** - Integración perfecta
5. **5 minutos de setup** - Más rápido que debuggear el problema actual



## Pasos Inmediatos

```
# 1. Crear cuenta en Neon.tech
open https://neon.tech

# 2. Crear proyecto
# 3. Copiar connection string (incluye pooling automático)

# 4. Ejecutar migración
cd /home/ubuntu/rental_management
export OLD_DB="postgresql://
role_43c1c84ad:sEELkXUYGfVvh4naABX9p2XZgDP3UKMC@db-43c1c84ad.db003.hosteddb.reai.io:
5432/43c1c84ad"
export NEW_DB="tu_nueva_url_de_neon"

# 5. Backup y restore
pg_dump "$OLD_DB" | psql "$NEW_DB"

# 6. Actualizar Vercel
echo "$NEW_DB" | npx vercel env add DATABASE_URL production

# 7. Deploy
npx vercel --prod
```



## Comparativa de Soluciones

Solución	Tiempo	Complejidad	Costo	Mantenimiento	Recomendado
<b>Neon.tech</b>	10 min	Baja	\$0	Bajo	★★★★★
<b>Supabase</b>	15 min	Baja	\$0	Bajo	★★★★★
<b>Railway</b>	10 min	Baja	\$5/mes	Bajo	★★★★★
Prisma Proxy	30 min	Media	\$25/mes	Medio	★★
Cloudflare Tunnel	2 horas	Alta	\$0	Alto	★
Servidor Proxy	4 horas	Muy Alta	\$10-20/mes	Muy Alto	★
Contactar Abacus	?	Baja	\$0	?	★



## Cambios Realizados

- ✓ Actualizado `lib/prisma.ts` con optimizaciones serverless

2.  Actualizado `DATABASE_URL` en Vercel con parámetros optimizados
  3.  Commit: "Optimize Prisma Client for serverless environment"
  4.  Redeploy a producción (2 veces con `-force`)
  5.  Diagnóstico completo del problema
- 

## URLs Útiles

- **Deployment actual:** <https://rental-management-4tgqq9mmw-jovaiks-projects.vercel.app>
  - **Neon.tech:** <https://neon.tech>
  - **Supabase:** <https://supabase.com>
  - **Railway:** <https://railway.app>
  - **Prisma Accelerate:** <https://www.prisma.io/data-platform/accelerate>
- 

## Próximos Pasos Sugeridos

1. **Decidir** qué solución implementar (recomiendo Neon.tech)
  2. **Crear cuenta** en el proveedor elegido
  3. **Migrar datos** desde Abacus.AI DB
  4. **Actualizar variables** de entorno en Vercel
  5. **Redeploy** y verificar funcionamiento
  6. **(Opcional)** Mantener backup en Abacus.AI DB si es necesario
- 

**Fecha:** 28 de Noviembre de 2025

**Proyecto:** Rental Management

**Status:**  Bloqueado por limitación de red (BD privada)

**Solución:**  Migrar a base de datos pública (Neon.tech recomendado)