

# Andy 21

## MariaDB 10.11 rompe WordPress en Plesk en Plesk



Actualizar MariaDB de 10.5 a 10.11 en Plesk dejó WordPress sin conexión a la base de datos. El culpable: SELinux. Aquí está el diagnóstico, la solución y lo que no debe repetirse.

Esta es la crónica de una mañana que empezó con una actualización rutinaria y acabó con un bug de SELinux, dos reinicios de servidor y la confirmación de que **la IA es ya un copiloto indispensable en el trabajo técnico.**

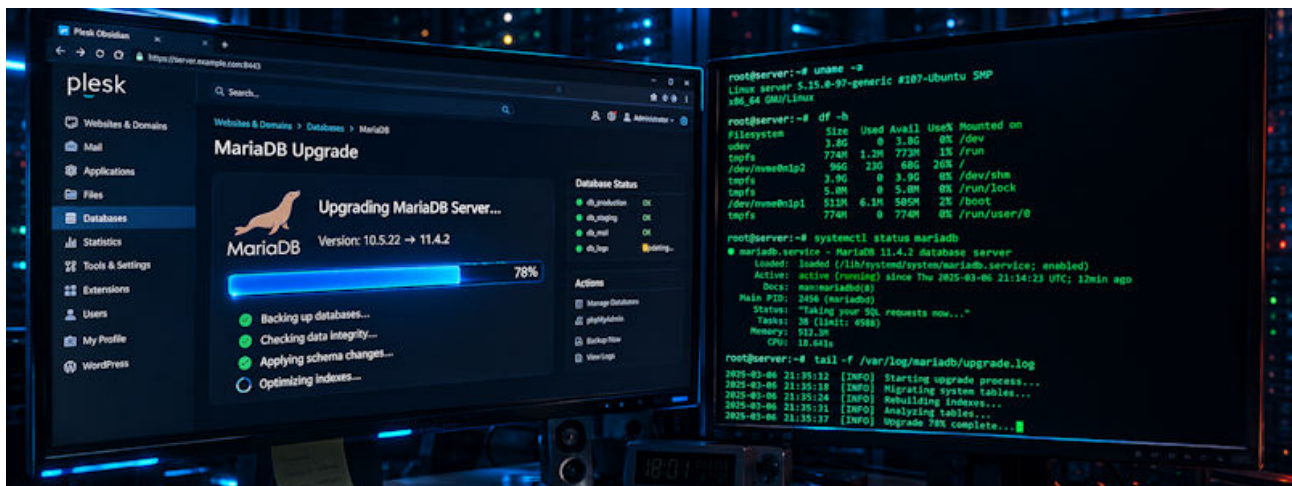
## ¿Por qué actualizar el servidor de bases de datos ahora?

WordPress 7.0 llega el **20 de mayo de 2026** e incorpora novedades importantes:

- Edición colaborativa en tiempo real (Fase 3 de Gutenberg)
- Cliente de IA integrado
- Rediseño del administrador con DataViews
- Revisiones visuales de bloques

Para aprovecharlas, la versión recomendada de base de datos es **MariaDB 10.6 o superior**. El servidor que aloja andy21.com corría MariaDB 10.5.29, una versión sin soporte activo desde hace tiempo. La actualización a 10.11 LTS (con soporte hasta febrero de 2028) era una deuda técnica pendiente.

*Antes de tocar los servidores de los clientes, se actualiza el propio. Si algo sale mal, mejor que sea en casa.*



## El proceso de actualización

La actualización se lanzó desde el panel Plesk, seleccionando la opción de actualizar el servidor de base de datos de 10.5.29 a 10.11. El proceso incluye una fase previa de backup de todas las bases de datos del servidor.

Y ahí apareció el primer obstáculo: la tabla `horde.horde_tokens` (el webmail de Plesk, apenas utilizado) llevaba años acumulando registros sin limpiar. El backup de esa tabla tardó más de media hora. No era un error, solo el coste de no haberla mantenido.

Mientras tanto, la instalación de MariaDB 10.11.16 completó sin problemas:

- MariaDB-server-10.11.16 instalado ✓

- Nuevo servicio activado ✓
- Actualización de estructura de tablas completada ✓

Pero al intentar acceder al backend de WordPress, apareció el temido:

*Error establishing a database connection*

## El diagnóstico: descartando causas

Con la ayuda de **Claude** (la IA de Anthropic), se fue descartando causa por causa de forma sistemática:

- MariaDB estaba corriendo correctamente (versión 10.11.16 activa)
- Las bases de datos existían y eran accesibles
- Los usuarios tenían permisos correctos
- Las credenciales del wp-config.php eran válidas
- La conexión manual por terminal funcionaba perfectamente
- PHP-FPM estaba activo y con los pools configurados

Todo apuntaba a que el problema era de comunicación entre PHP-FPM y MariaDB, no de configuración. Pero ¿por qué algunos sitios del mismo servidor funcionaban y otros no?

*Cuando todo parece correcto y nada funciona, el problema suele estar donde menos se busca.*

## La causa: SELinux y un bug conocido

Una búsqueda en los foros oficiales de Plesk dio con la respuesta. Es un **bug documentado** que afecta a servidores con AlmaLinux 9 al actualizar MariaDB de 10.5 a 10.11 mediante el actualizador de Plesk:

**SELinux bloquea el acceso de PHP-FPM al socket Unix de MariaDB** porque el binario `/usr/sbin/mariadb` queda con un contexto SELinux incorrecto tras la actualización. Los paquetes de MariaDB del repositorio oficial no incluyen las políticas SELinux correctas para AlmaLinux 9. El socket afectado es `/var/lib/mysql/mysql.sock`

Fuente: [Foro oficial de Plesk](#) y [base de conocimiento de Plesk](#).

# La solución: dos comandos

La solución es restaurar el contexto SELinux correcto para el binario de MariaDB y reiniciar el servicio:

```
semanage fcontext -a -t mysqld_exec_t /usr/sbin/mariadb  
restorecon /usr/sbin/mariadb  
systemctl restart mariadb
```

Tras ejecutar estos tres comandos, todos los sitios volvieron a funcionar de inmediato.

## Lecciones aprendidas

- **Horde/webmail:** si no se usa, conviene limpiar periódicamente la tabla `horde_tokens` para que no ralentice futuras operaciones de backup o actualización
- **SELinux:** en AlmaLinux 9, cualquier actualización de MariaDB desde el repositorio oficial de MariaDB puede dejar el contexto SELinux incorrecto. Es el primer sitio donde mirar si PHP no puede conectar al socket tras una actualización
- **IA como copiloto:** el diagnóstico sistemático junto a Claude permitió descartar causas rápidamente y encontrar la solución correcta en los foros especializados. Sin ese apoyo, probablemente el proceso habría llevado el doble de tiempo o más
- **Primero en casa:** probar en el servidor propio antes que en los de clientes no es precaución excesiva, es el protocolo correcto de cualquier profesional experimentado

El proceso queda ahora documentado para replicarlo en los servidores de clientes con la misma configuración, sin sorpresas.

---

Si te ha gustado este PDF, compártelo. Si no te ha gustado, díselo a [Andy](#).