

Privilegios que se pueden otorgar a los usuarios:

Sólo el usuario 'root' posee todos los derechos disponibles.

Este rol actúa como plantillas que facilitan la asignación de permisos a un usuario de una base de datos. Se soportan los siguientes roles de usuario de base de datos: Lectura y escritura (usado de forma predeterminada), Solo lectura y Sólo escritura. Cada rol presenta un conjunto de privilegios predeterminados que se conceden al usuario de la base de datos al que se le asigne.

Otros privilegios que se pueden conceder son:

- **ALL:** para conceder todos los privilegios.
- **CREATE:** permite crear nuevas tablas.
- **DELETE:** permite usar la sentencia DELETE.
- **DROP:** permite borrar tablas.
- **INSERT:** permite insertar datos en tablas.
- **UPDATE:** permite usar la sentencia UPDATE.

Los privilegios se agrupan de manera lógica de acuerdo con el área del privilegio.

- ★ Privilegios FILE: los privilegios que comienzan con la cadena file funcionan en los objetos del sistema de archivos. Por ejemplo, el privilegio file_dac_write anula el control de acceso discrecional al escribir en los archivos.
- ★ Privilegios IPC: los privilegios que comienzan con la cadena ipc anulan los controles de acceso a objetos IPC. Por ejemplo, el privilegio ipc_dac_read permite a un proceso leer memoria compartida remota que está protegida por DAC.
- ★ Privilegios NET: los privilegios que comienzan con la cadena net otorgan acceso a funcionalidades de red específicas. Por ejemplo, el privilegio net_rawaccess permite a un dispositivo conectarse con la red.
- ★ Privilegios PROC: los privilegios que comienzan con la cadena proc permiten a los procesos modificar propiedades restringidas del propio proceso. Los privilegios PROC incluyen privilegios que tienen un efecto muy limitado. Por ejemplo, el privilegio proc_clock_highres permite a un proceso usar temporizadores de alta resolución.
- ★ Privilegios SYS: los privilegios que comienzan con la cadena sys otorgan a los procesos acceso sin restricciones a distintas propiedades del sistema. Por ejemplo, el privilegio sys_linkdir permite a un proceso establecer y anular enlaces físicos a directorios.

En MySQL existen cinco niveles distintos de privilegios:

- Globales: se aplican al conjunto de todas las bases de datos en un servidor. Es el nivel más alto de privilegio, en el sentido de que su ámbito es el más general.

- De base de datos: se refieren a bases de datos individuales, y por extensión, a todos los objetos que contiene cada base de datos.
- De tabla: se aplican a tablas individuales, y por lo tanto, a todas las columnas de esas tabla.
- De columna: se aplican a una columna en una tabla concreta.
- De rutina: se aplican a los procedimientos almacenados. Aún no hemos visto nada sobre este tema, pero en MySQL se pueden almacenar procedimientos consistentes en varias consultas SQL.

Modelo jerárquico:

Todo conjunto de datos, almacenados en gran cantidad, requiere de un sistema de gestión tanto el almacenamiento como el acceso de dicha información. Un SGBD (Sistema de Gestión de Base de Datos) es el encargado de esta tarea y lo hace de varias maneras diferentes.

Los SGBD jerárquicos fueron los primeros en aparecer. Una base de datos jerárquica se puede visualizar como una estructura en árbol. Las bases de datos jerárquica son bastantes rígidas, una vez diseñada la base de datos, es compleja cambiarla y además, es necesario un conocimiento amplio de la forma en la que se han almacenado los datos para poder superarlos de forma efectiva.

Los SGBD jerárquicos son modelos lógicos basados en registros que se usan para describir datos en los niveles lógicos y de vista, se llaman así debido a que la base de datos se estructura en registro de formato fijo de diferentes tipos, en cada tipo de registro se define un número fijo de campos o atributos y cada campo tiene normalmente una longitud fija. El uso de registros de longitud fija simplifica la implementación en el nivel físico de la base de datos.

Concepto:

Es un tipo de sistema de gestión de base de datos, almacena información en una estructura jerárquica que enlaza los registros en forma de estructura de árbol (similar a un árbol visto al revés), en donde un nodo padre de información puede tener varios nodos hijos y así sucesivamente.

El modelo jerárquico no diferencia una vista lógica de una vista física de la base de datos. De manera que las relaciones entre datos se establecen siempre a nivel físico, es decir, mediante referencia a direcciones físicas del medio de almacenamiento (sectores y pistas).

Los datos se almacenan en la forma de registro, el equivalente a las tuplas del modelo relacional. Cada registro consta de un conjunto de campos, el equivalente a los tributos del modelo relacional. Un conjunto de registro con los mismos campos se denomina fichero (record type, en inglés), el equivalente a las relaciones del modelo relacional.

Modelo de red:

En este modelo las entidades se representan como nodos y sus relaciones son las líneas que los unen. En esta estructura cualquier componente puede relacionarse con cualquier otro. El Modelo de Red se puede entender como una extensión del modelo jerárquico. También se presenta mediante un árbol, pero en este caso, cada

hijo puede tener varios padres. De este modo se reducen, o eliminan, las redundancias, Pero desaparece la herencia de los campos. La integridad de datos, asociada a los arcos padre-hijo, se mantiene.

Una Base de Datos de Red se compone de dos conjuntos:

- El Conjunto de los Registros. Un conjunto de instancias múltiples de varios tipos de registros.
- El Conjunto de las Relaciones. Un conjunto de instancias múltiples de varios tipos de relaciones.