UNIVERSIDAD DEL AZUAY

Facultad de Ciencias de la Administración
Escuela de Ingeniería de Sistemas

"Estudio práctico sobre la seguridad de los datos con el gestor de base de datos Microsoft Sql Server 2000"

Trabajo de graduación previo a la obtención del título de "Ingeniero de Sistemas"

Autores: Pedro Rodas Palomeque
Lolita Ulloa Brito

Director: Ing. Oswaldo Merchán

Cuenca, Ecuador
2006

Dedicatoria

Este trabajo lo dedico a mi familia, quienes me han brindado su ayuda incondicional, para lograr una de las metas de mi vida, terminar mi carrera profesional.

LOLITA

Dedicatoria

Dedico este estudio a mis padres, hermanos, y amigos que, con su dedicación y bondad, me han apoyado cada día para sacar adelante mis trabajos e intenciones, y así poder alcanzar mis metas y sueños.

PEDRO

AGRADECIMIENTO

A Dios, por darnos fuerza, sabiduría, seguridad y firmeza para realizar con éxito el presente trabajo de graduación.

Agradecemos de manera especial al Ing. Oswaldo Merchán, director de la presente monografía, por su asesoramiento y apoyo, el cuál fue de gran ayuda para la realización de este trabajo.

Nuestro sincero agradecimiento a la “Universidad del Azuay” y a sus profesores que, por su enseñanza, han podido contribuir en la formación académica y personal en nuestros días de vida estudiantil

INDICE DE CONTENIDOS

INDICE DE ILUSTRACIONES Y CUADROS

INDICE DE ANEXOS

RESUMEN

Actualmente se conoce de las constantes amenazas que tienen las bases de datos dentro de una empresa, por varios motivos como: la vulnerabilidad en sus gestores, conocimiento reducido por parte del administrador de base de datos en cuanto a seguridad, usuarios malintencionados, o ataques de manipulación o destrucción de la información. En base a una investigación profunda encontramos que la mejor solución a esta problemática es plantear un estudio teórico-práctico que fortalezca la seguridad de los datos, proponiendo soluciones a la no autorizada manipulación de la información.

Toda base de datos debe poseer un plan de seguridad consistente para controlar que actividades se pueden desarrollar y que información debe ser restringida. Mediante los estudios realizados podemos garantizar que un sistema de seguridad sólido protege los datos, independientemente de la manera en que los usuarios accedan a la información. La investigación se basará en los mecanismos e implementación de seguridad para la interfaz de usuario y objetos de la base de datos. En este proyecto se estudian los diferentes temas de seguridad como: arquitectura de seguridad, diseño de la seguridad, creación de cuentas de seguridad, administración de cuentas de seguridad, administración de permisos, temas avanzados de seguridad, para la protección de la información frente a peligros. Para este propósito se utilizará el gestor de base de datos Microsoft Sql Server 2000, utilizando un caso práctico en donde se implementan las diferentes maneras de concesión y revocación de permisos para un usuario.

Al culminar este proyecto se observa una reducción considerable de los problemas de pérdida de datos, y del acceso no autorizado de usuarios, contribuyendo a certificar la información y aportando al mantenimiento del servidor con un óptimo nivel de seguridad.

ABSTRACT

At the moment we know about that the databases within a company are under constant threat due to several reasons such as the vulnerability of their managers, the database administrator’s little knowledge regarding security, malicious users, handling attacks, or destruction of information. Based on a deep investigation, we found that the best solution to this problem is to raise a theoretical-practical study that strengthens the safety of the data by proposing solutions to unlicensed manipulation of information.

All databases should have a consistent security plan to control what activities can be developed and what information should be restricted. By means of the studies carried out, we can guarantee that a solid security system protects the data independently of the way the users accede to the information. The research will be based upon the mechanisms and implementation of security for user's interface and objects of the database. This project covers different topics on security, such as security architecture, security design, creation of security accounts, administration of security accounts, licenses administration, and advanced topics about security for the protection of information in case of risks. For this purpose the Manager Microsoft Sql Server 2000 database will be used in practical situation where the different ways for concession and revocation of a user’s licenses are implemented.

By the end of this project, a considerable reduction of problems concerning both the loss of data and the unlicensed access of users is observed, which contributes to certify the information and the maintenance of the server with an optimal security level.

INTRODUCCION

Este trabajo es una aplicación de los conocimientos adquiridos durante nuestros estudios en el área de bases de datos, el mismo que es de gran importancia para nuestra vida profesional. Mediante este podemos aportar soluciones para la seguridad de los datos.

En la actualidad, al hablar de informática y de sistemas, es imposible no tocar el tema de las bases de datos, las mismas que han ganado extensa importancia dentro de las empresas, ayudando a organizar de una manera óptima la información, considerando que esta es de vital importancia para la toma de decisiones y el funcionamiento adecuado de la organización.

Como es conocido, son constantes las vulnerabilidades y amenazas que sufren estos depósitos de datos, produciéndose esto por varias razones, como son: reducido conocimiento en la administración de las bases de datos por parte de sus encargados, fortaleza insuficiente en la seguridad brindada por los diferentes gestores de bases de datos del mercado, y frecuente manipulación de los datos por parte de personas ajenas a los mismos, que con fines fraudulentos pretenden afectar a la empresa.

Esta monografía nos da a conocer a modo práctico la arquitectura, diseño, creación y administración de cuentas, administración de permisos y temas avanzados de seguridad, para la correcta protección de los datos y para su adecuado manejo.

La realización de este trabajo permitirá determinar mecanismos de seguridad óptimos para el acceso eficiente a los datos, desarrollar una aplicación que permita ilustrar los mecanismos de seguridad del gestor, además ayudará a especificar responsabilidades a los diferentes usuarios.

CAPITULO 1: SEGURIDAD EN LOS DATOS

Introducción

Al hablar de seguridad, implica comprender la aplicación y emplear las medidas de seguridad adecuadas en el entorno de trabajo. Se deber desarrollar seguridades que identifiquen que usuarios pueden acceder a determinados datos y en las base de datos que actividades pueden desarrollar. Es fundamental que solo las personas autorizadas tengan acceso a los datos, de lo contrario es seguro encontrar a alguien destruyendo datos, o accediendo a información privada. Por medio de este capítulo podremos conceptualmente aprender los mecanismos de seguridad como cuentas, autenticación, roles, permisos de seguridad y creación de vistas, a modo de introducción.

1.1 Cuentas de seguridad

Existen dos tipos de cuentas de seguridad. El primero garantiza el acceso al servidor de base de datos, el cual se denomina cuenta de seguridad de inicio de sesión. El segundo garantiza el acceso a una base de datos dentro de un servidor, el cual se denomina cuenta de seguridad de usuario. Las cuentas de seguridad permiten que los usuarios de una aplicación tengan acceso a los recursos creados para su utilización. Las cuentas de inicio de sesión pueden poseer varias cuentas de seguridad asociadas, una por cada base de datos a la que acceda el inicio de sesión.[1]

Las cuentas de inicio de sesión pueden ser de varias formas, como son:

- Usuario del Sistema Operativo.
- Grupo del Sistema Operativo.
- Estándar.

1.2 Autenticación

La autenticación es el proceso mediante el cual un servidor de bases de datos acepta la cuenta de inicio de sesión, determinando su validez. Existen dos modos de autenticación, estos son:

- Autenticación de modo mixto

[1] DOBSON. Rick. Programación de Microsoft SQL SERVER 2000 con Microsoft VISUAL BASIC.NET. España: Madrid, 2002. Pág. 234-235.

- Autenticación modo sistema operativo

Al ejecutar con el modo de autenticación del sistema operativo, los usuarios de la base de datos no requieren validarse cuando estos desean acceder al servidor. De lo contrario, al ejecutar el modo mixto, únicamente, los usuarios de bases de datos con inicio de sesión del gestor, necesariamente enviarán su nombre de inicio de sesión y contraseña, cuando intenten ingresar. El gestor crea automáticamente un inicio de sesión para la cuenta administrador, disfrutando prácticamente de todos los privilegios de la cuenta tradicional de seguridad "sa". Se debe considerar que el inicio de sesión de la cuenta administrador puede ser eliminado, pero el inicio de sesión "sa" (predeterminado) no puede ser borrado.[2]

1.3 Roles y permisos

Estos son privilegios que permiten a los usuarios accionar sobre los objetos de la base de datos. Los objetos de la base de datos son elementos que pueden ser aplicados a la protección de la seguridad. Al asignar un rol a un inicio de sesión, y a una cuenta de seguridad de usuario, se otorgan permisos para desarrollar determinadas tareas, tanto sobre el servidor, como en la base de datos. Además de los permisos otorgados directamente a los usuarios, se puede asignar permisos mediante roles. Existen dos colecciones de roles fijos (roles establecidos por el gestor), los del servidor y los de bases de datos. También los gestores permiten la creación de roles por parte de los usuarios: con estos se pueden asignar permisos precisos, necesarios para una aplicación. Los permisos son de dos tipos generales. Primero, pueden asignarse permisos sobre objetos de bases de datos, denominados "permisos de objetos". Segundo, se puede garantizar la autoridad necesaria para desarrollar un grupo de sentencias, creando permisos para su ejecución, a estos se les denomina "permisos de sentencia".[3]

1.4. Características especiales de la seguridad

1.4.1 Creación de Vistas

Además de las restricciones de acceso a las tablas proporcionadas por los privilegios

[2] DOBSON. Dic. Programación de Microsoft SQL SERVER 2000 con Microsoft VISUAL BASIC.NET. España: Madrid, 2002. Pág. 235-236.

[3] Libros en pantalla de SQL Server. Versión 2000.

de SQL, las vistas también desempeñan un papel fundamental en la seguridad de SQL. Al definir una vista, y otorgando a los usuarios permisos para su acceso, se permite seleccionar únicamente ciertas filas y columnas de una tabla. Con esto, se consigue restringir la información, parcializando su contenido. Las vistas pueden combinar datos de una o más tablas, proporcionando exactamente los datos que necesita un usuario concreto, y negándole el acceso al resto de los datos. Las vistas no se pueden utilizar de manera constante para restringir el acceso a la base de datos, sin hacer que esta sufra problemas de rendimiento. [4]

1.5 Conclusiones

Al finalizar este capítulo podemos darnos cuenta de las diversas alternativas que tenemos para proteger nuestros datos. Mediante este estudio, se mencionan ciertas pautas de seguridad como es la confidencialidad, integridad y disponibilidad de los datos.

[4] KORTH. Henry. SILBERSCHATZ. Abraham. SUDARSHAN. S. Fundamentos de bases de datos. Editorial Mc Graw-Hill. 3edición. España: Madrid, 1998. Pág. 73-74

CAPITULO 2: ASIGNACION DE ROLES A LOS USUARIOS

Introducción

Los roles controlan el acceso a los servidores y objetos dentro de una base de datos, manejando permisos. Un rol es un grupo al que los inicios de sesión y usuarios individuales pueden ser adicionados, para que los permisos puedan ser aplicados a un grupo, en lugar de aplicar los permisos a todos los usuarios en forma individual, con este concepto se estudiara los roles de servidor, base de datos, aplicación y usuario descritos a continuación.

2.1 Roles de servidor

Los roles de servidor son utilizados para otorgar privilegios de seguridad en todo el servidor a un inicio de sesión, tienen asociados permisos que sirven para realizar tareas, como la creación, modificación, borrado de una base de datos, o la administración de inicios de sesión de inicios de otros usuarios y la modificación de sus contraseñas.

Un inicio de sesión puede pertenecer a uno, ninguno, o más de uno de estos roles. Los usuarios de bases de datos poseen la capacidad de heredar los permisos de cualquier rol fijo de servidor al que pertenezca su inicio de sesión. No se puede asignar una cuenta de usuario directamente a un rol fijo de servidor. [1]

ROL	*DESCRIPCION*
Sysadmin	Realiza cualquier tarea y tiene acceso sin restricción a todas las base de datos
Serveradmin	Configura opciones globales del servidor.
Setupadmin	Designa procedimientos almacenados para ejecutarse al inicio y administra servidores vinculados.
Securityadmin	Administra los inicios de sesión del servidor, los permisos de creación de base de datos y cambio de contraseñas.
Processadmin	Administra procesos que se ejecutan en le gestor.
Dbcreator	Crea y modifica las bases de datos.
Diskadmin	Administra los archivos de disco.
Bulkadmin	Puede ejecutar una operación de inserción masiva.

[1] Tabla 2.1: Roles fijos de servidor.

[1] Libros en pantalla de SQL Server. Versión 2000.

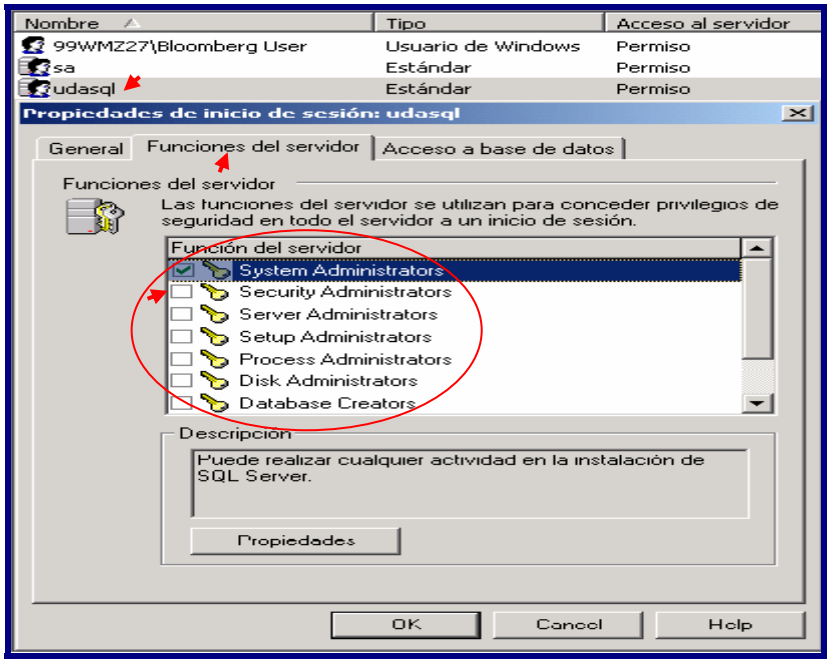


Figura 2.1: Asignación de roles de servidor utilizando interfaz gráfica.

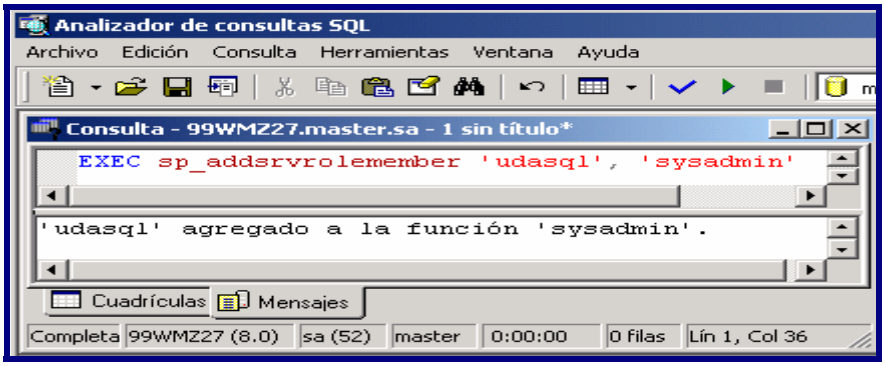


Figura 2.2: Asignación de roles de servidor para un inicio de sesión Sql utilizando el analizador de consulta.

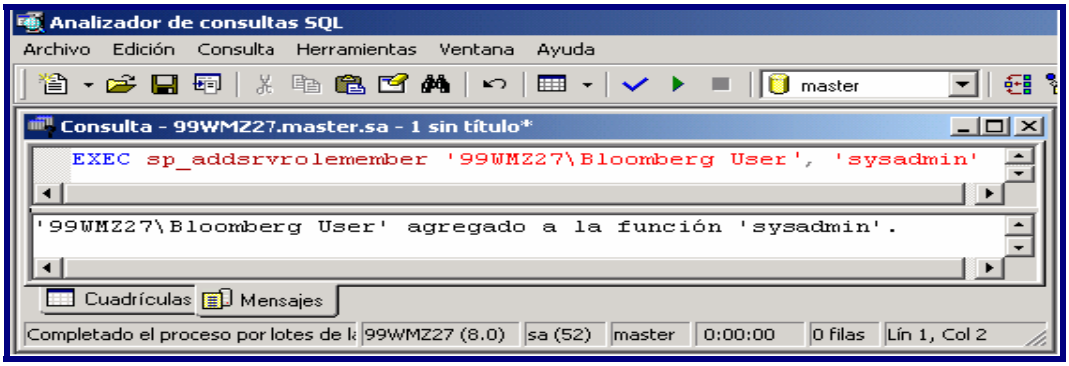


Figura 2.3: Asignación de roles de servidor para un inicio de sesión Windows utilizando el analizador de consulta.

Para ejecutar el procedimiento sp_addsrvrolemember se debe ser miembros de la función fija de servidor sysadmin. Los miembros de una función fija de servidor pueden ejecutar sp_addsrvrolemember para agregar miembros solo a la misma función fija de servidor.

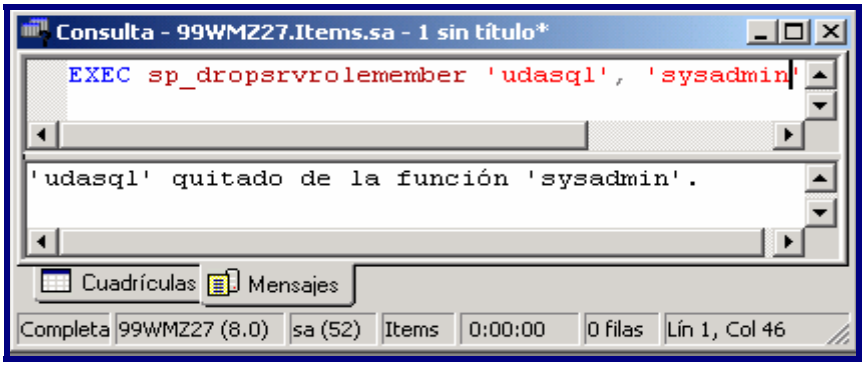


Figura 2.4: Borrado de roles de servidor para un inicio se sesión Sql utilizando el analizador de consulta.

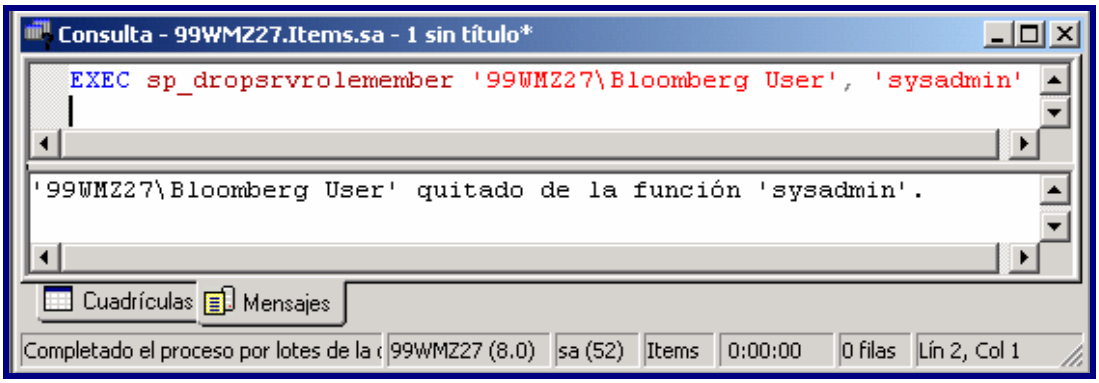


Figura 2.5: Borrado de roles de servidor para un inicio de sesión Windows utilizando el analizador de consulta.

Los miembros de la función fija de servidor sysadmin pueden ejecutar sp_dropsrvrolemember para quitar inicios de sesión de una función fija de servidor. Los miembros de una función fija de servidor pueden quitar a otros miembros de la misma función fija de servidor.

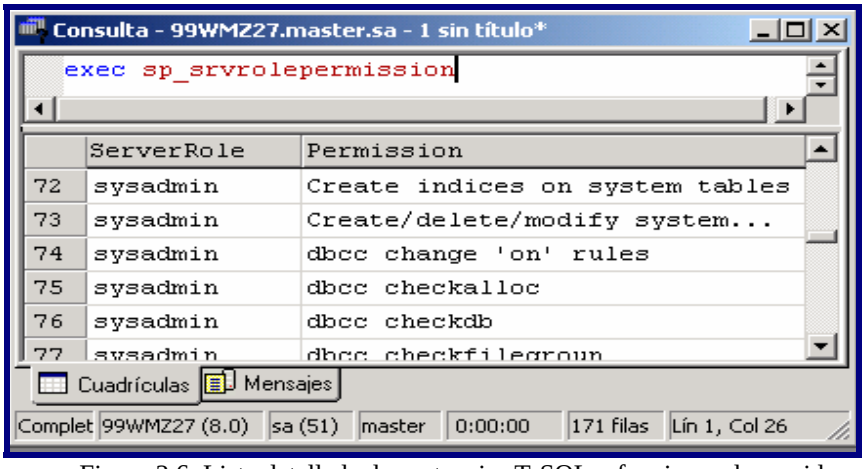


Figura 2.6: Lista detallada de sentencias T-SQL y funciones de servidor a las que tiene acceso cada rol de servidor.

Los permisos de ejecución para el procedimiento sp_srvrolepermission corresponden a la función public.

2.2 Roles de base de datos

Estos roles conceden a los usuarios de una base de datos capacidad de consultar o modificar, así como añadir nuevos objetos de la base de datos. La cuenta de seguridad utilizada para designar estos roles puede ser una cuenta de usuario de la base de datos actual basada en un inicio de sesión del gestor o un inicio de sesión del sistema operativo. Aparte de los roles fijos de una base de datos existe otro rol dentro de cada base de datos: el rol public (público). Todos los usuarios de base de datos pertenecen al rol public y heredan cualquier permiso de este rol. La pertenencia a un rol en una base de datos no otorga la pertenencia al mismo rol para cualquier otra base de datos. Además, el rol public de una base de datos puede poseer diferentes permisos que el rol public de otra base de datos. Una cuenta de usuario de modo invitado (guest), permite el acceso a una base de datos con cualquier inicio de sesión, ya que pertenece al rol public, por este motivo resulta una buena práctica eliminar todos los permisos del rol public cuando se desea enfatizar en la seguridad.[2]

ROL	***DESCRIPCION***
db_owner	Realiza cualquier tarea sobre una base de datos.
db_accessadmin	Ejecuta procedimientos sp_grantdbaccess y sp_revokedbaccess.
db_securityadmin	Ejecuta procedimientos sp_addrolemember y sp_droprolemember.
db_ddladmin	Ejecuta sentencias CREATE, ALTER y DROP sobre los objetos de una base de datos.
db_backupoperator	Realiza operaciones BACKUP DATABASE y BACKUP LOG.
db_datareader	Realiza operaciones SELECT en cualquier objeto de una base de datos.
db_datawriter	Ejecuta operaciones INSERT, UPDATE, Y DELETE sobre cualquier objeto de una base de datos
db_denydatareader	Niega operaciones SELECT en cualquier objeto de una base de datos.
db_denydatawriter	Niega operaciones INSERT, UPDATE, Y DELETE sobre cualquier objeto de una base de datos

[2] Tabla 2.2: Roles fijos de base de datos.

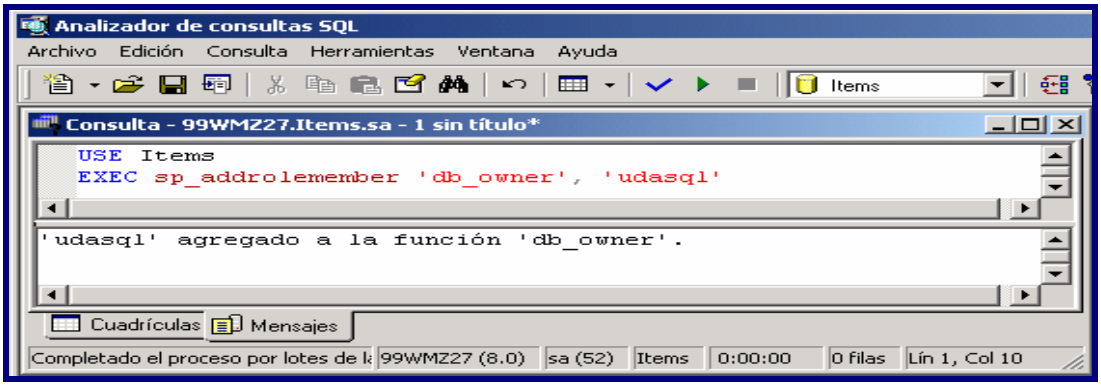


Figura 2.7: Asignación de roles de base de datos para un usuario Sql utilizando el analizador de consulta.

[2] Libros en pantalla de SQL Server. Versión 2000.

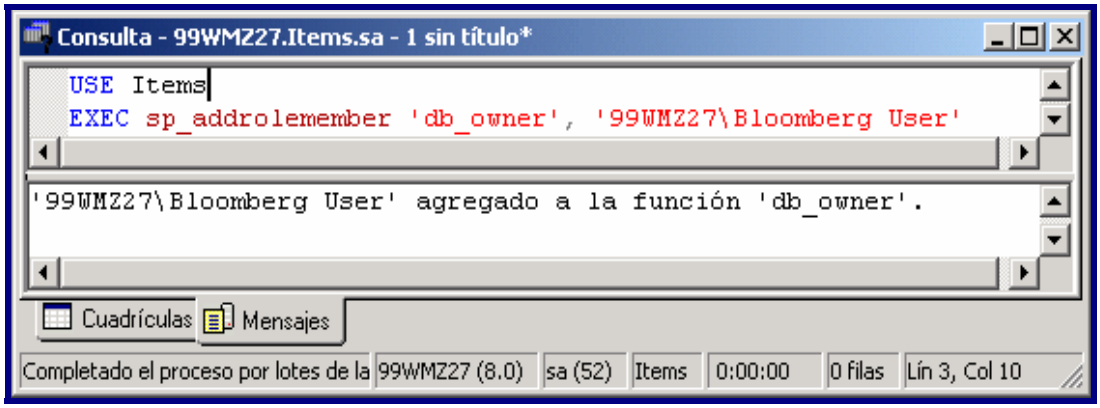


Figura 2.8: Asignación de roles de base de datos para un usuario Windows utilizando el analizador de consulta.

Unicamente los miembros de la función fija de servidor sysadmin y la función fija de base de datos db_owner pueden ejecutar sp_addrolemember.

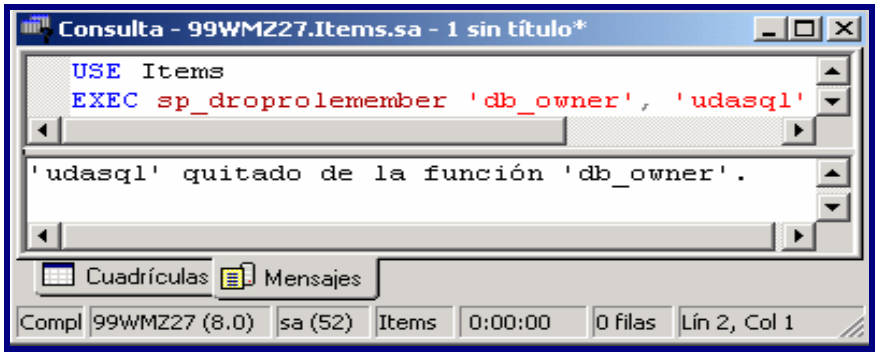


Figura 2.9: Borrado de roles de base de datos para un inicio se sesión Sql utilizando el analizador de consulta.

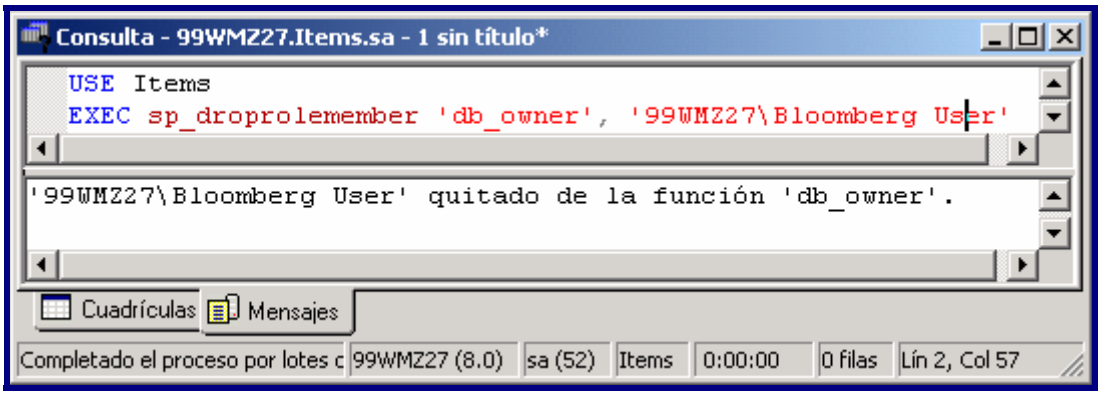


Figura 2.10: Borrado de roles de base de datos para un inicio de sesión Windows utilizando el analizador de consulta.

Los miembros de la función sysadmin o de las funciones de base de datos db_securityadmin y db_owner pueden ejecutar sp_droprolemember. Unicamente los miembros de la función de base de datos db_owner pueden quitar usuarios de una función fija de base de datos.

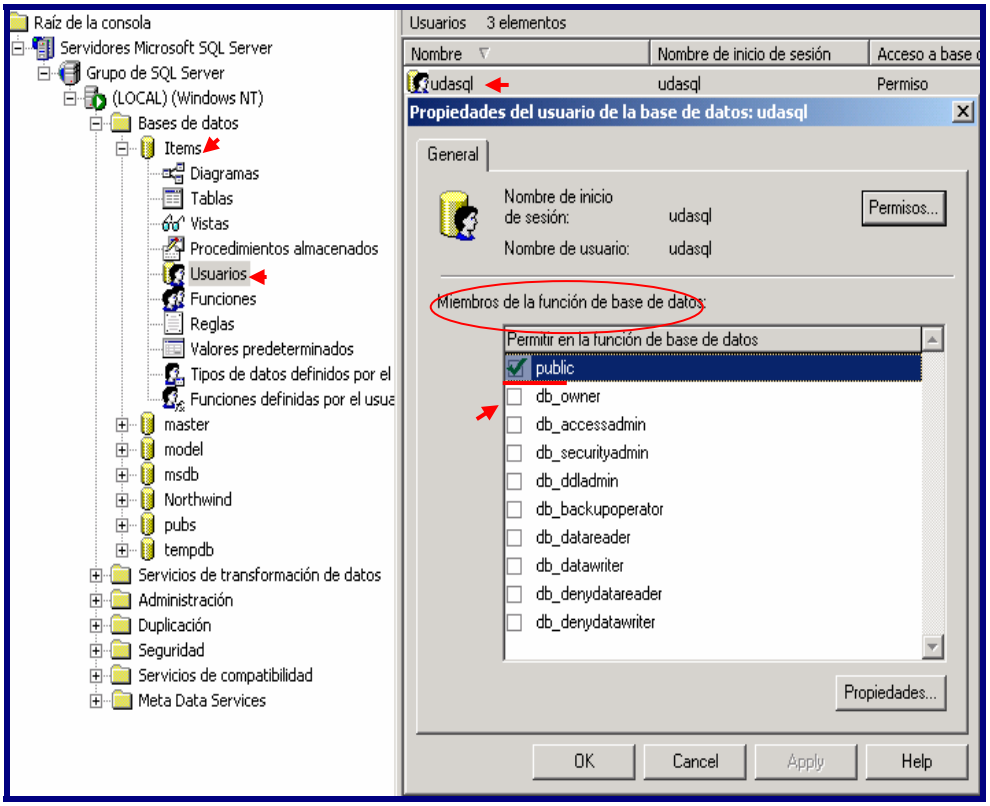


Figura 2.11: Asignación de roles de base de datos utilizando interfaz gráfica.

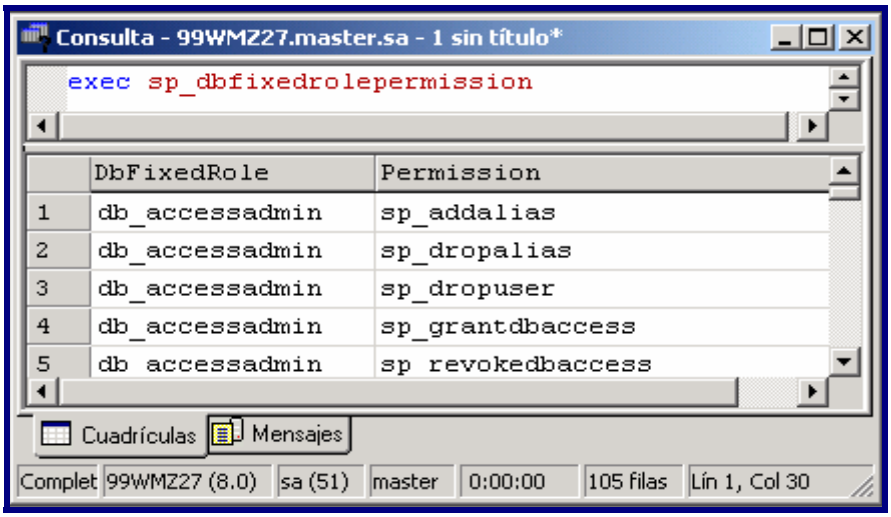


Figura 2.12: Lista de permisos asociados con cada uno de los roles fijos de base de datos.

Todos los usuarios tienen permiso para ejecutar sp_dbfixedrolepermission.

2.3 Roles de aplicación

Los roles de aplicación simplifican el trabajo de los administradores de base de datos, debido a que estos no estarían preocupados en manejar los permisos a nivel de usuario individual. Todo lo que necesitan hacer es crear un rol de aplicación y asignar permisos a este rol. La aplicación que está conectándose a la base de datos

activa el rol de aplicación y hereda los permisos asociados con este rol, perdiendo el resto de parámetros de seguridad asociados con una conexión a una base de datos, por esto el usuario debe desconectarse y volverse a conectar con un inicio de sesión y cuenta de usuario, para poseer nuevamente los parámetros de seguridad. Un usuario de un rol de aplicación disfruta de todos los permisos de un usuario guest en la base de datos actual o en cualquier base de datos. Alguna de las características de los roles de aplicación son los siguientes:

- No existen roles de aplicación pre-definidos.
- Los roles de aplicación necesitan ser activados en tiempo de ejecución, por la aplicación, usando una contraseña.
- Los roles de aplicación sobrescriben los permisos estándar.
- Los roles de aplicación son específicos a la base de datos.

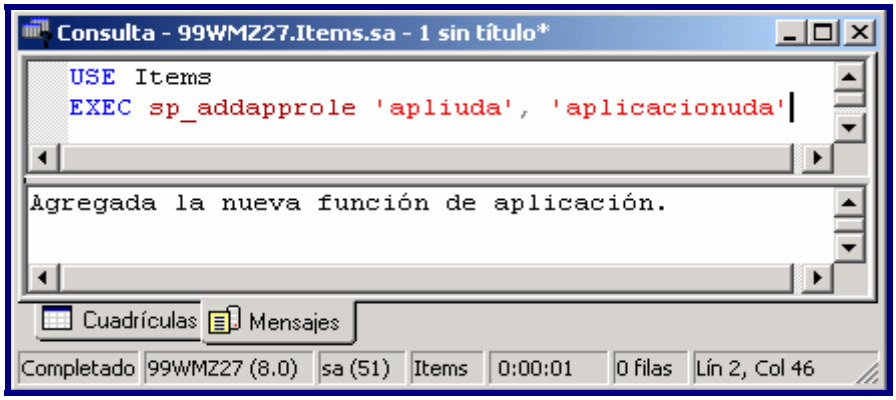


Figura 2.13: Creación de un rol de aplicación, por medio del analizador de consulta.

Unicamente los miembros de la función de servidor sysadmin y de las funciones de base de datos db_securityadmin y db_owner pueden ejecutar sp_addapprole.

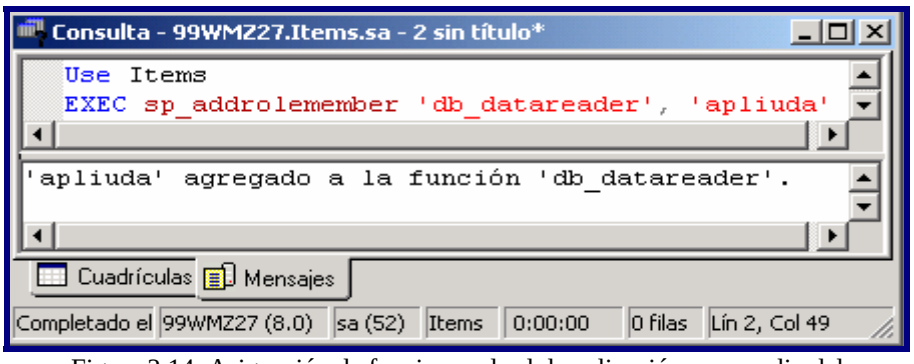


Figura 2.14: Asignación de funciones al rol de aplicación, por medio del analizador de consulta.

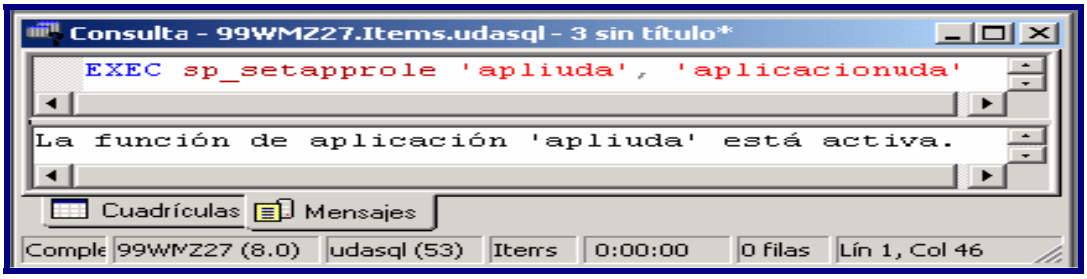


Figura 2.15: Registro de los usuarios en el rol de aplicación, activando los permisos asociados a este, en la base de datos actual, por medio del analizador de consulta.

Todos los usuarios logran ejecutar sp_setapprole si proveen la contraseña correcta de la función de aplicación.

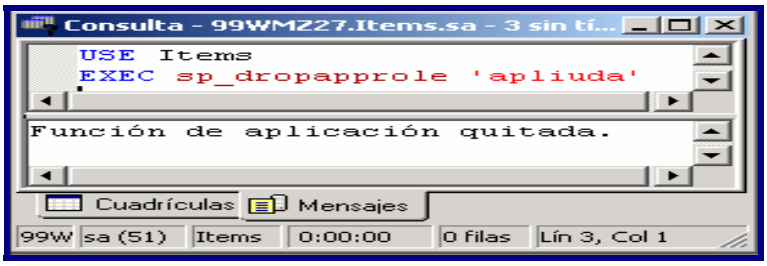


Figura 2.16: Quita una función de aplicación de la base de datos actual, por medio del analizador de consulta.

Los miembros de la función sysadmin o de las funciones de base de datos db_securityadmin y db_owner logran ejecutar sp_dropapprole.

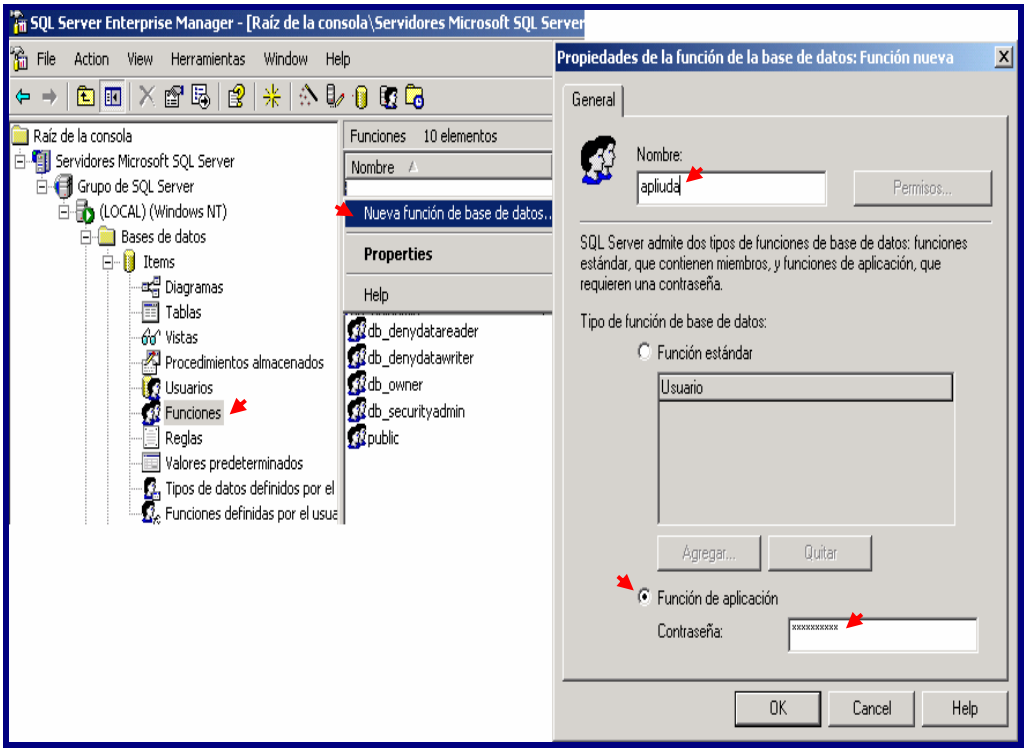


Figura 2.17: Creación de un rol de aplicación, por medio de interfaz gráfica.

Para eliminar un rol de aplicación por medio de la interfaz gráfica escoja un servidor, expanda base de datos y a continuación expanda la base de datos en la que se encuentra la función de aplicación. Luego de clic en funciones, escoja la función que desea eliminar y luego escoja la opción eliminar.

2.4 Roles definidos por usuario

Se crean estos roles cuando se necesita asignar permisos de forma más granular que los proporcionados por los roles fijos. Dos roles definidos por el usuario en diferentes base de datos con el mismo nombre pueden tener miembros diferentes y conceder permisos diferentes dentro de cada base de datos. Se pueden administrar los roles definidos por el usuario con cualquier inicio de sesión que pertenezca a un rol de servidor sysadmin o a un rol de base de datos db_owner o db_securityadmin.

Se debe tomar en consideración los permisos de ejecución de sentencias SQL que se pueden añadir a los roles definidos por el usuario. Con estos permisos los miembros de los roles definidos por el usuario pueden realizar tareas tales como: crear bases de datos, tablas, vistas, procedimientos almacenados, y funciones definidas por el usuario sobre una base de datos.

SQL ofrece tres sentencias para administrar los permisos que un rol puede conceder a sus miembros, estas son: GRANT, REVOKE, y DENY. Con la sentencia GRANT se otorga permisos a un rol. REVOKE y DENY inhabilitan permisos a un rol. Un permiso obtenido por medio de la sentencia GRANT, sobrescribe una sentencia REVOKE para el mismo permiso en cualquier otro rol. DENY sobrescribe cualquier sentencia GRANT.

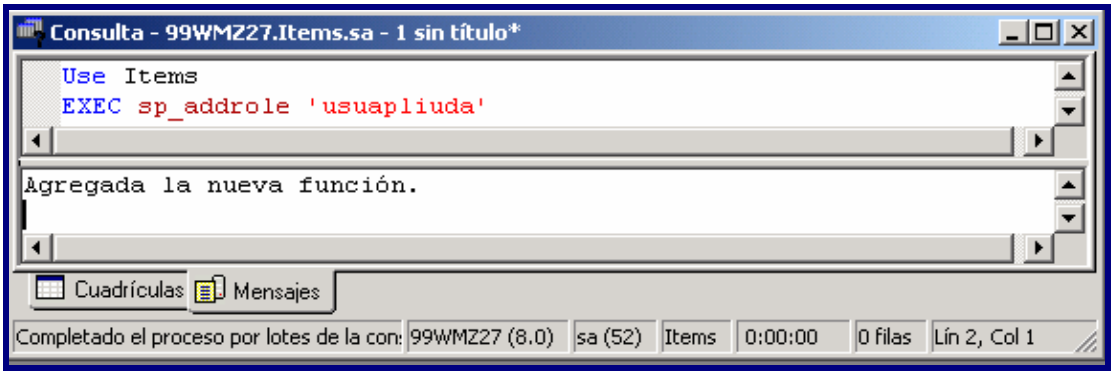


Figura 2.18: Creación de un rol defenido por el usuario, utilizando el analizador de consulta.

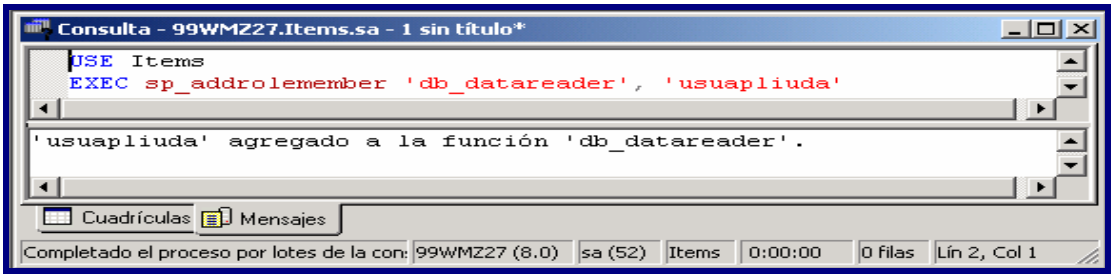


Figura 2.19: Asignación de funciones al rol definido por el usuario , utilizando el analizador de consulta.

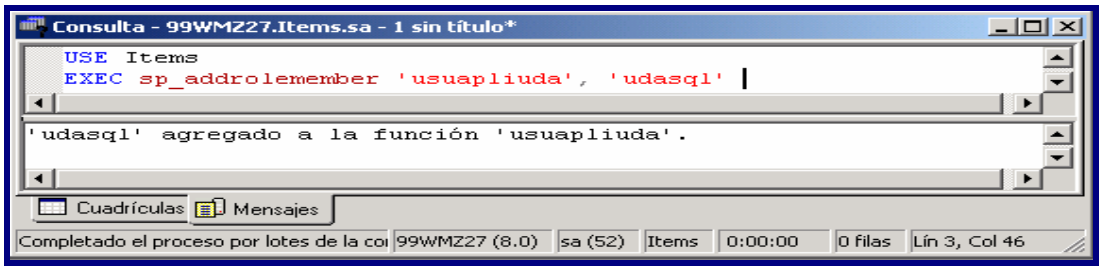


Figura 2.20: Registro de miembros en el rol definido por el usuario, por medio del analizador de consulta.

Nombre del usuario sin colocar el nombre del servidor

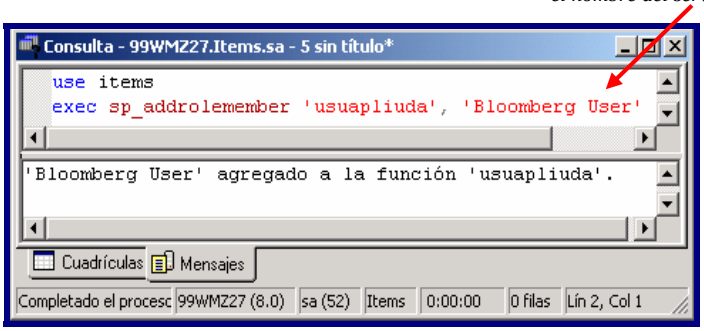


Figura 2.21: Registro de miembros Windows en el rol definido por el usuario, por medio del analizador de consulta.

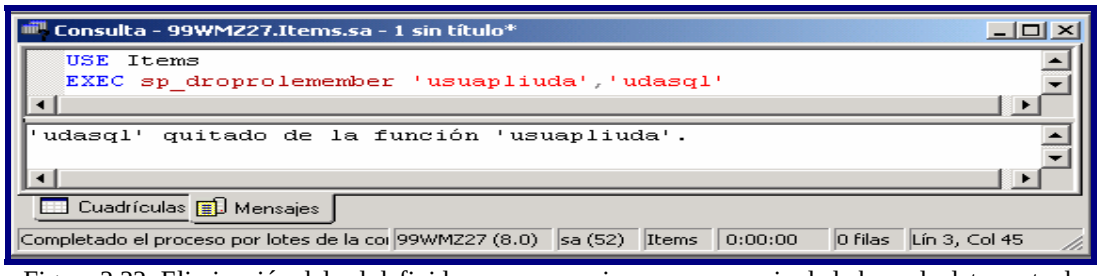


Figura 2.22: Eliminación del rol definido por un usuario para un usuario de la base de datos actual, utilizando el analizador de consulta.

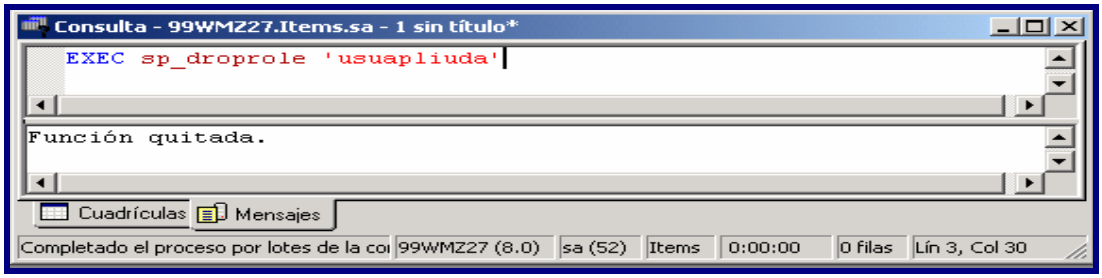


Figura 2.23: Eliminación del rol definido por un usuario, utilizando el analizador de consulta.

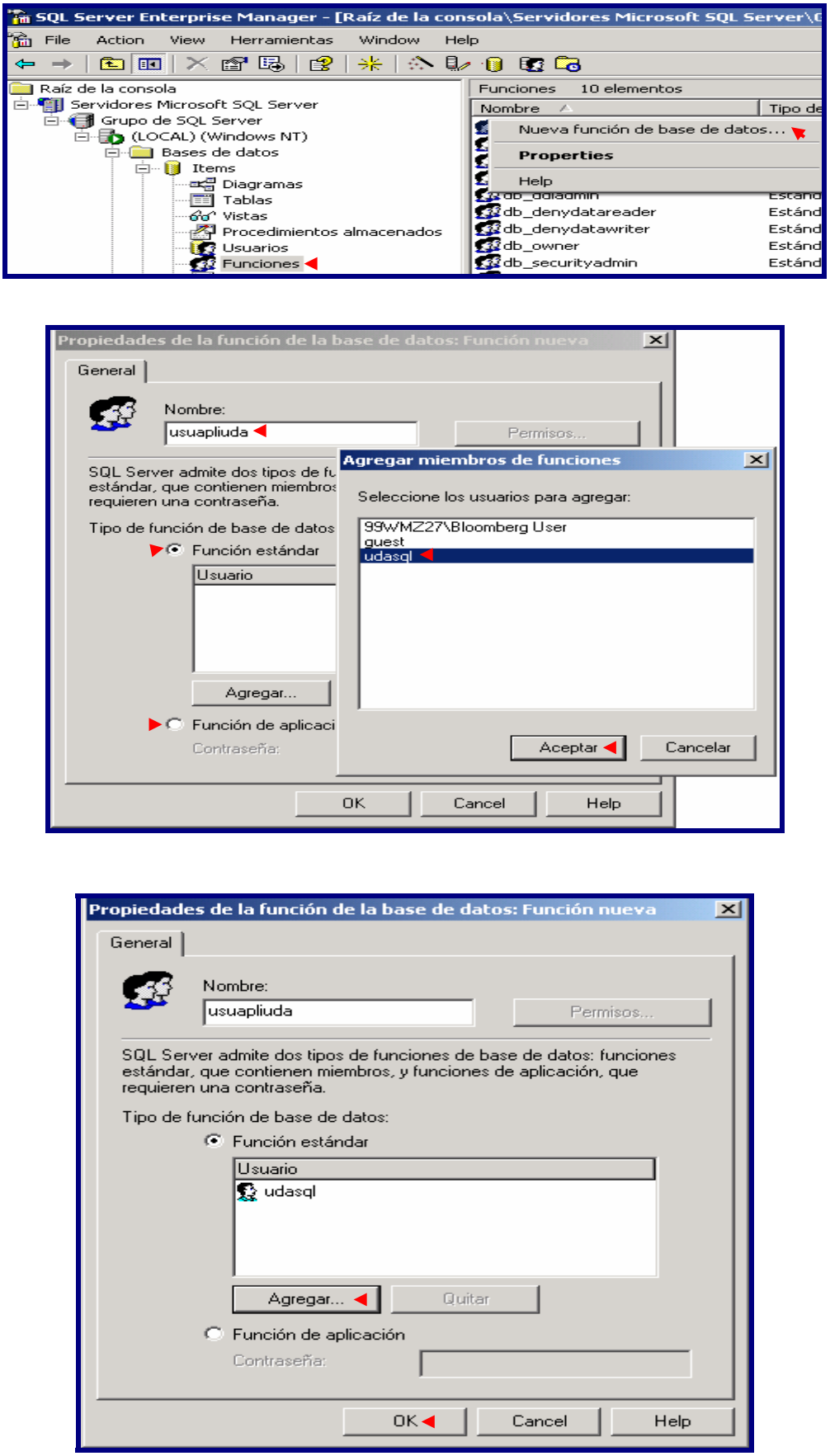


Figura 2.24: Creación de un rol definido por el usuario, asignando miembros a este, por medio de interfaz gráfica.

Se pueden asignar privilegios a los roles definidos por el usuario, permitiendo aplicar los permisos SELECT, INSERT, UPDATE y DELETE sobre cualquier objeto de una base de datos.

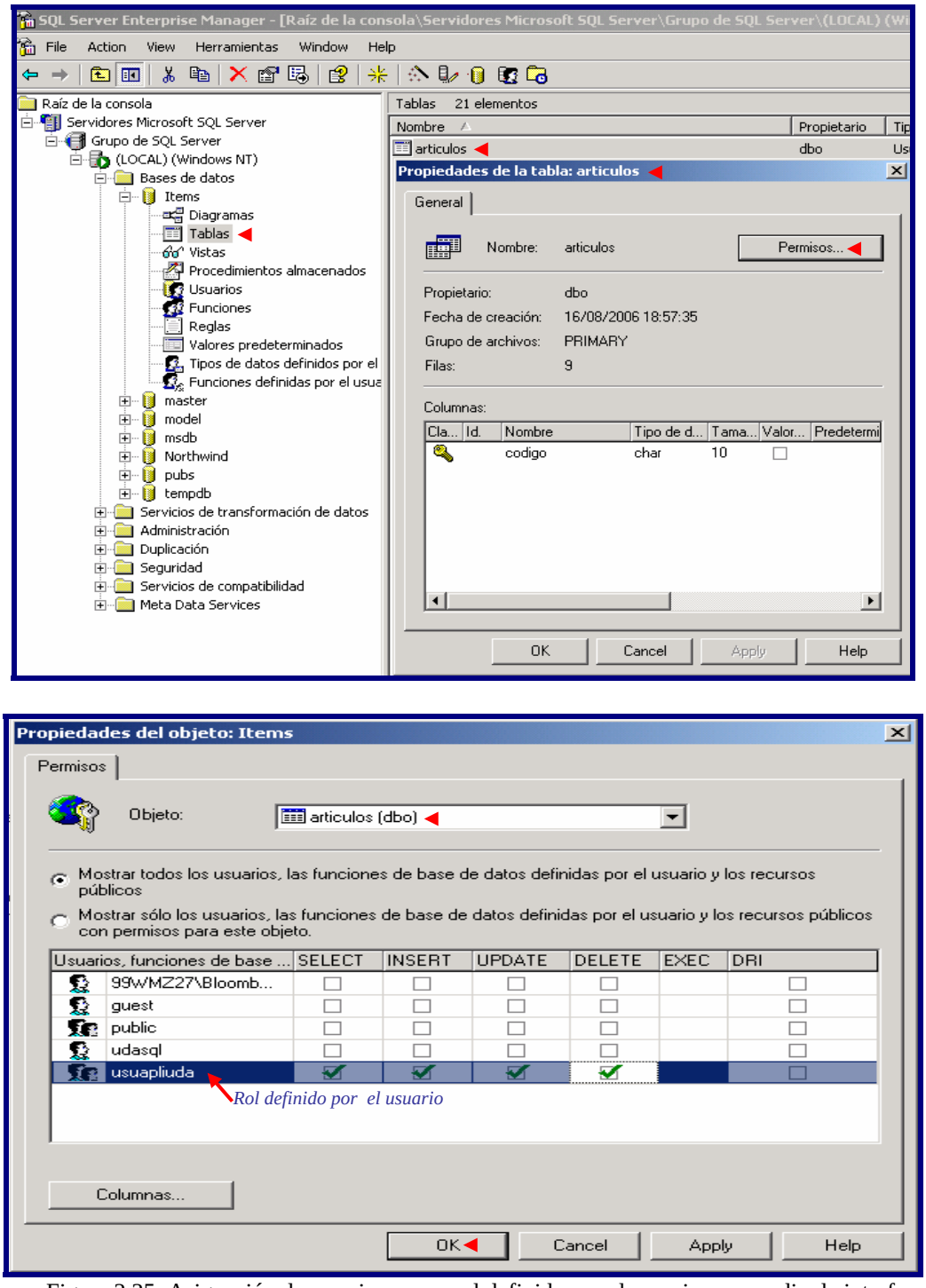


Figura 2.25: Asignación de permisos a un rol definido por el usuario, por medio de interfaz gráfica.

Para eliminar un rol definido por el usuario por medio de la interfaz gráfica escoja un servidor, expanda base de datos y a continuación expanda la base de datos en la que se encuentra la función definida por el usuario, de clic en funciones y escoja la función que desea eliminar y luego escoja la opción eliminar.

2.5 Conclusiones

Un punto crítico dentro de las protección de los datos es su seguridad, por lo que es importante que se conozca los conceptos y ejecución de roles aprendidos en este capítulo, ya que mediante estos podemos emplear roles de servidor y aplicación para un inicio de sesión, como roles de base de datos, aplicación y usuario para una cuenta de usuario. Además por medio de la asignación de roles y permisos se llega a la integridad de los datos, que consiste en que solo las personas autorizadas puedan variar (modificar o borrar) los datos, también se puede consolidar la disponibilidad, que se cumple si las personas autorizadas pueden acceder a tiempo a la información, fijando una disponibilidad adecuada a los datos, ya que tiene mucha importancia la disponibilidad absoluta.

CAPITULO 3: CREACION DE INICIO DE SESION Y DE USARIO

Introducción

Recuerde que los inicios de sesión y las cuentas de usuario se complementan entre sí, y que se pueden crear desde cualquier base de datos del servidor. En este capítulo se toca puntualmente a modo práctico los temas de creación y borrado de un inicio de sesión y usuario tanto para el gestor de base de datos como para Windows. Además se demuestra la utilización de los diversos componentes del gestor.

3.1 Componentes SQL Server

Antes de utilizar los componentes del gestor se debe cerciorar de que todos los servicios se encuentren activos.

3.1.1 Administrador de servicios

Es utilizado para iniciar, detener y pausar los componentes de Microsoft® SQL Server™ 2000 en el servidor.[1]

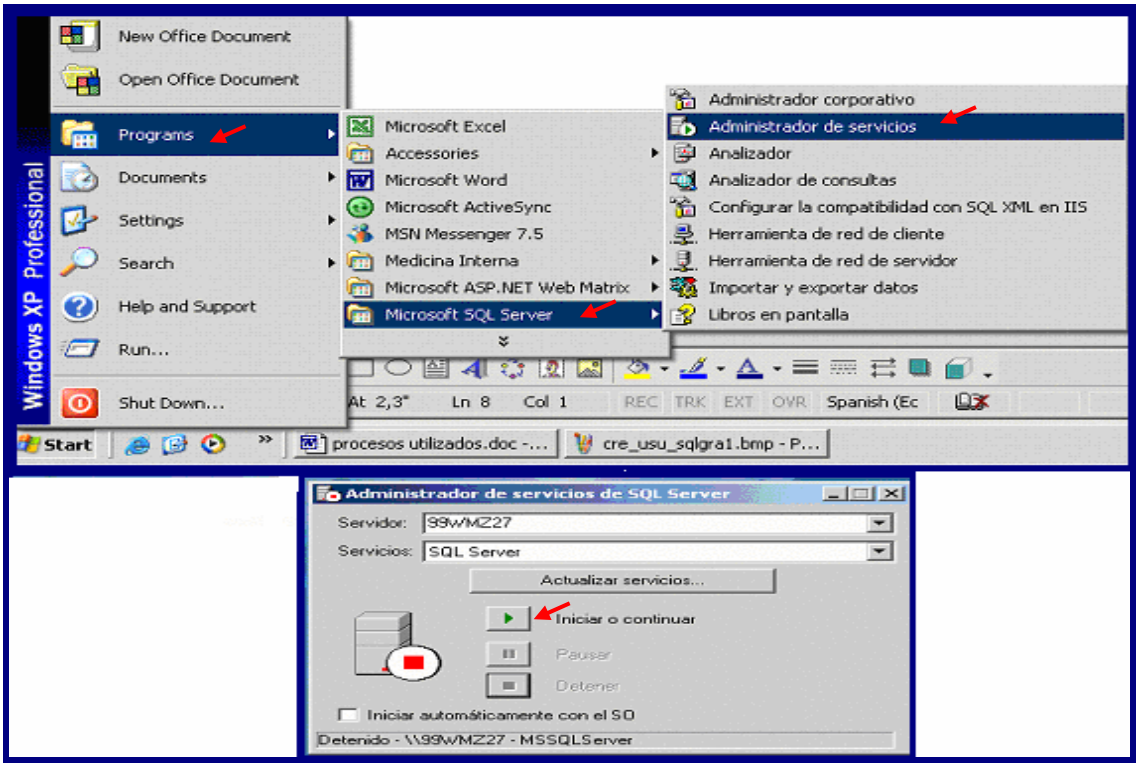


Figura 3.1: Administrador de servicios.

[1] Libros en pantalla de SQL Server. Versión 2000.

3.1.2 Administrador corporativo

Manipula objetos de la base de datos, verifica que se pueda acceder a una conexión SQL y WINDOWS o únicamente WINDOWS, además mediante este componente se debe comprobar la existencia de las bases de datos con las que se desea trabajar.[2]

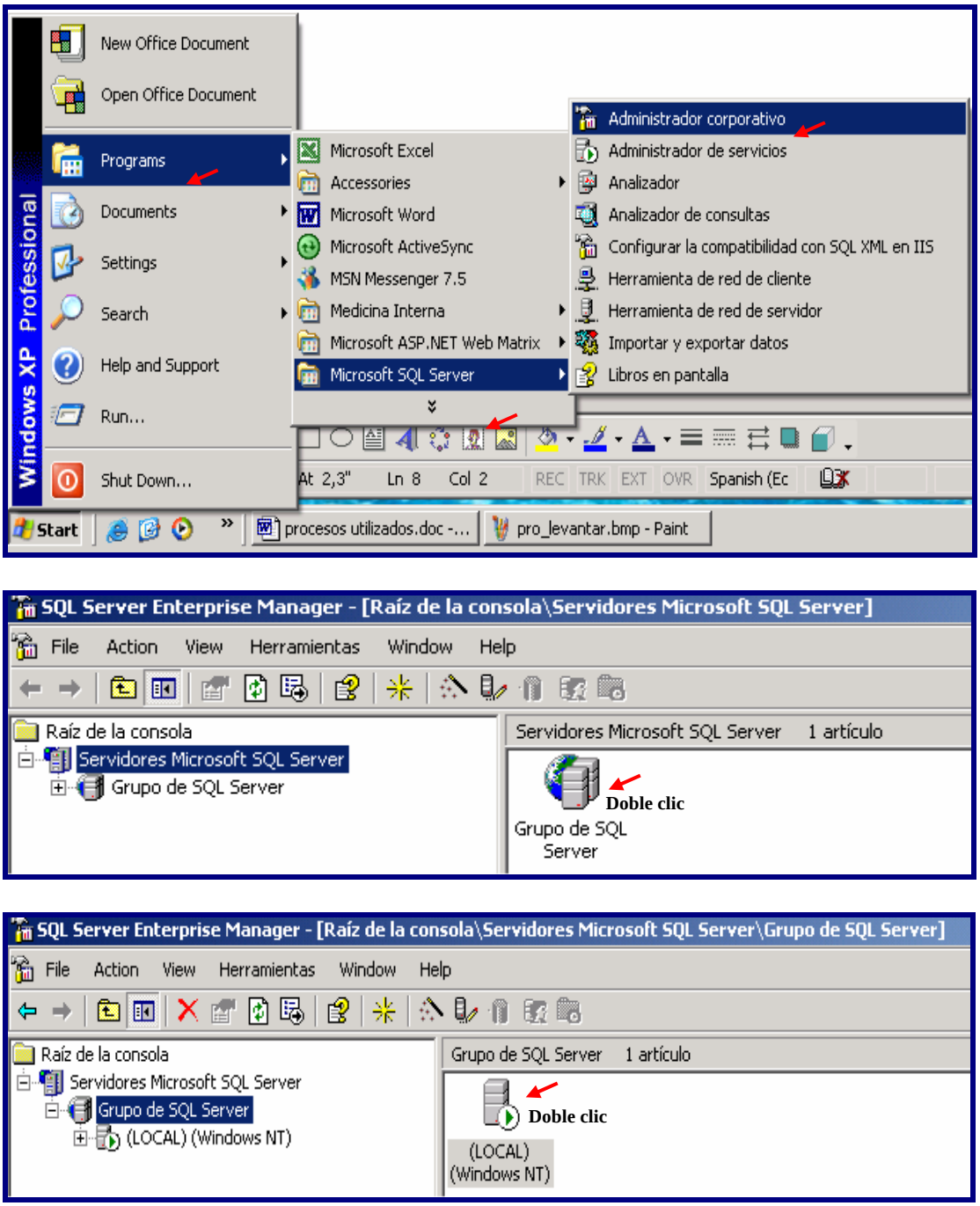


Figura 3.2: Administrador corporativo.

[2] Libros en pantalla de SQL Server. Versión 2000.

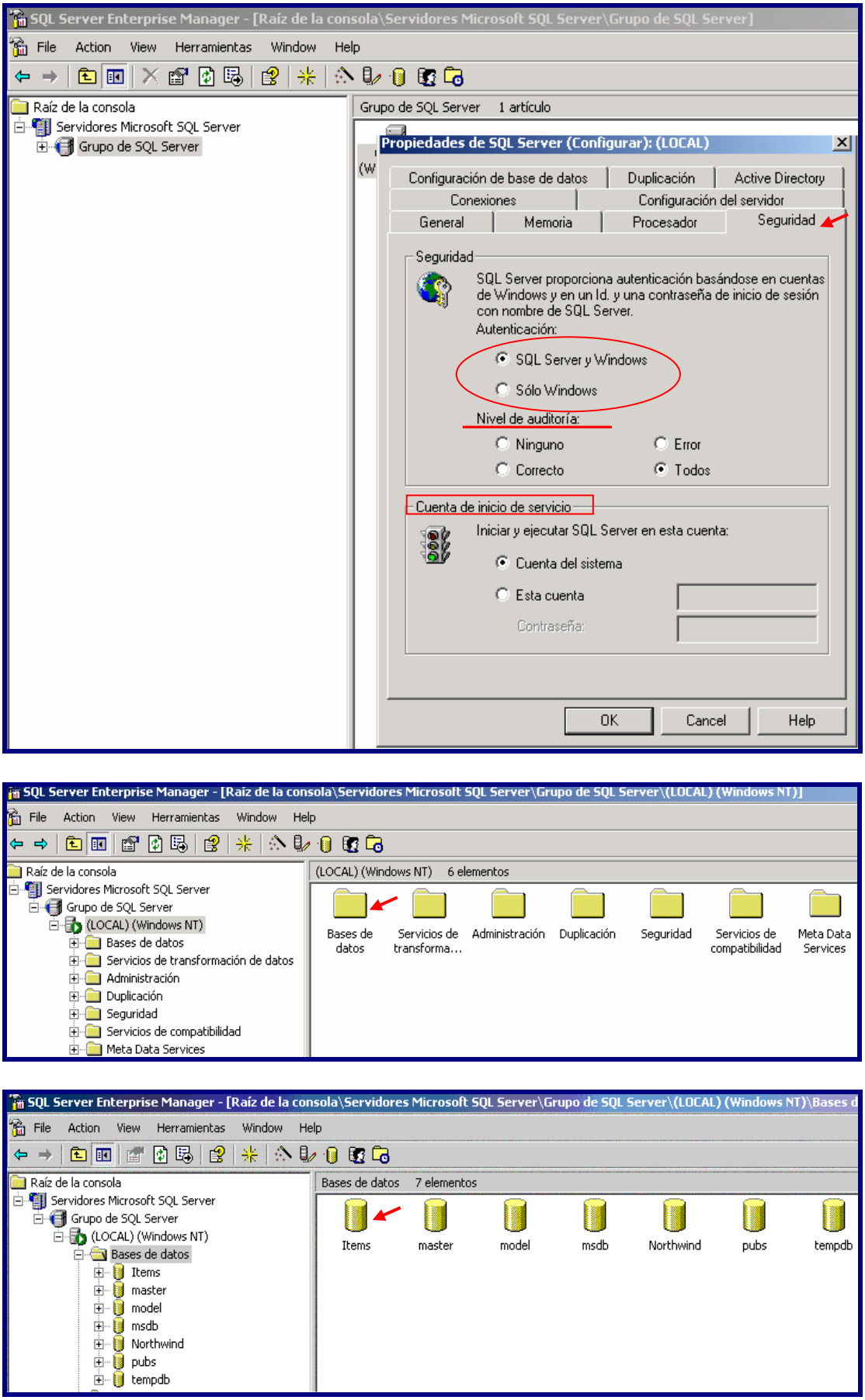


Figura 3.3: Seguridad en el administrador corporativo.

3.1.3 Analizador de consulta

Es una herramienta gráfica interactiva que ayuda a un administrador o a un operador de bases de datos a escribir, ejecutar simultáneamente consultas, ver los resultados, analizar el plan de consultas y recibir asistencia para mejorar el rendimiento de las consultas.[3]

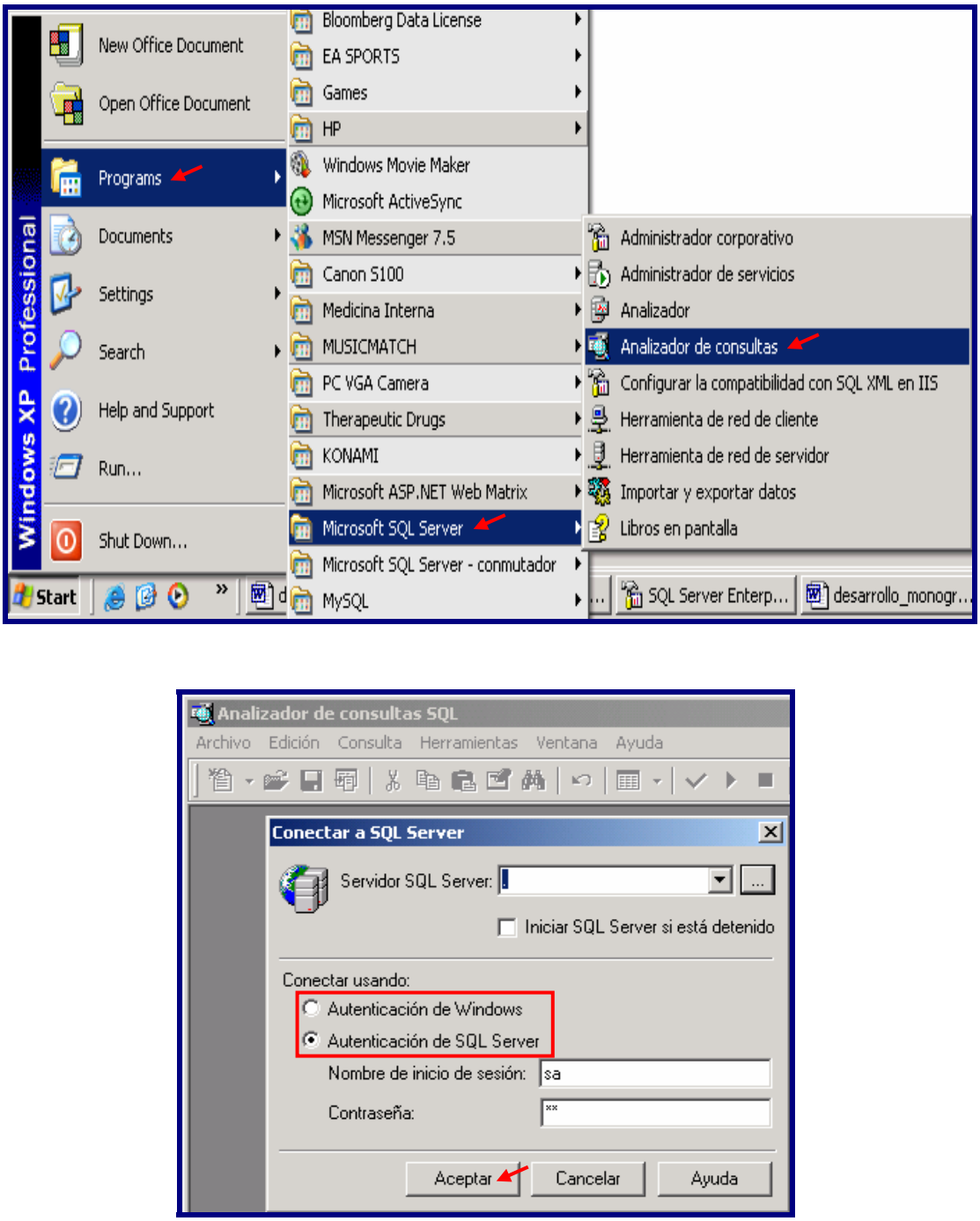


Figura 3.4: Analizador de consultas.

[3] Libros en pantalla de SQL Server. Versión 2000.

3.2 Creación de un inicio de sesión y usuario

Un inicio de sesión requiere una cuenta de seguridad para cada base de datos a la que va a acceder el usuario. Sin embargo, existen dos modos para que un usuario pueda acceder a una base de datos sin una cuenta de usuario: mediante la cuenta de invitado (guest), disfrutando de cualquier permiso asignado explícitamente a esta cuenta o indirectamente mediante el rol public de la base de datos, o de lo contrario con un inicio de sesión perteneciente al rol del servidor sysadmin, accediendo a cualquier base de datos del servidor, sin ninguna restricción en su funcionamiento; es por esto que se debe considerar la limitación de inicios de sesión que pertenezcan al rol sysadmin.

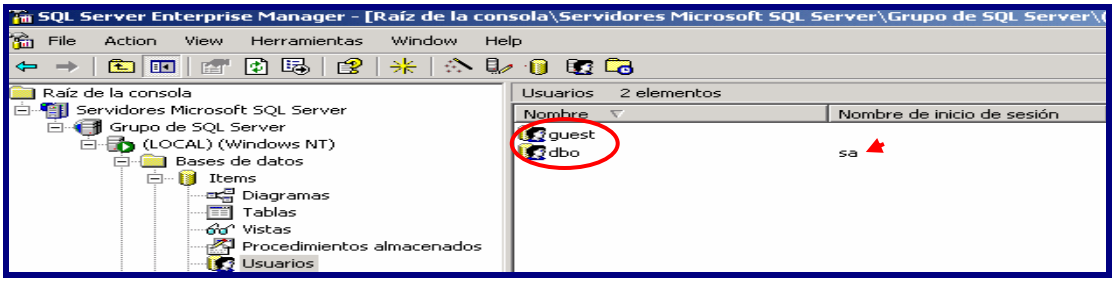


Figura 3.5: Cuenta de usuario invitado (guest), inicio de sesión del rol sysadmin.

Sí se requiere que un usuario de un inicio de sesión interactúe de una u otra forma con una base de datos específica, se deberá crear una cuenta de usuario para el inicio de sesión de la base de datos. Para crear un inicio de sesión se debe ser miembro del rol de servidor sysadmin o securityadmin como (la cuenta de seguridad tradicional sa) y ejecutar los siguientes procedimientos:

PROCEDIMIENTO	***DESCRIPCION***
sp_addlogin	Procedimiento almacenado del sistema, crea un inicio de sesión administrado por el gestor de base de datos, es obligatorio definir los argumentos @loginame y @passwd. Si no se asigna una base de datos al argumento @defdb, la base de datos predeterminada hubiera sido la base de datos master (Es una base de datos integrada que Sql Server utiliza para administrarse así mismo).
sp_password	Procedimiento almacenado del sistema, cambia contraseña de un usuario, únicamente los miembros del rol de servidor sysadmin y securityadmin pueden modificar una contraseña de un inicio de sesión diferente al suyo. Ejemplo EXEC sp_password 'uda2006', '2006uda' (Cambia la contraseña del inicio de sesión udasql)
sp_grantdbaccess	Procedimiento almacenado del sistema, crea una cuenta de usuario sobre un inicio de sesión en una base de datos. Unicamente los miembros del rol de servidor sysadmin así como los miembros de los roles de base de datos db_owner y db_accessadmin podrán ejecutar este procedimiento. Antes de ejecutar este procedimiento se debe estar seguro de la base de datos que se necesita utilizar para la creación de la cuenta de usuario, esto se lo puede hacer mediante la sentencia USE con el nombre de la base de datos.

Tabla 3.1: Procedimientos almacenados del sistema.

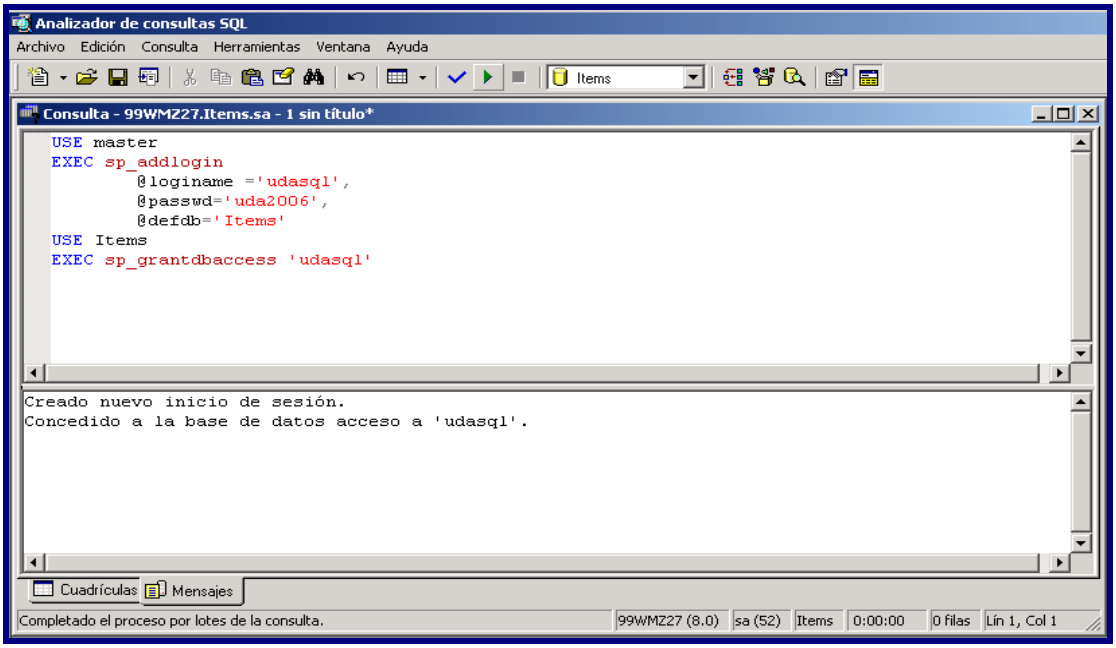


Figura 3.6: Crea un inicio de sesión SQL Server con acceso a la base de datos ítems mediante el analizador de consulta.

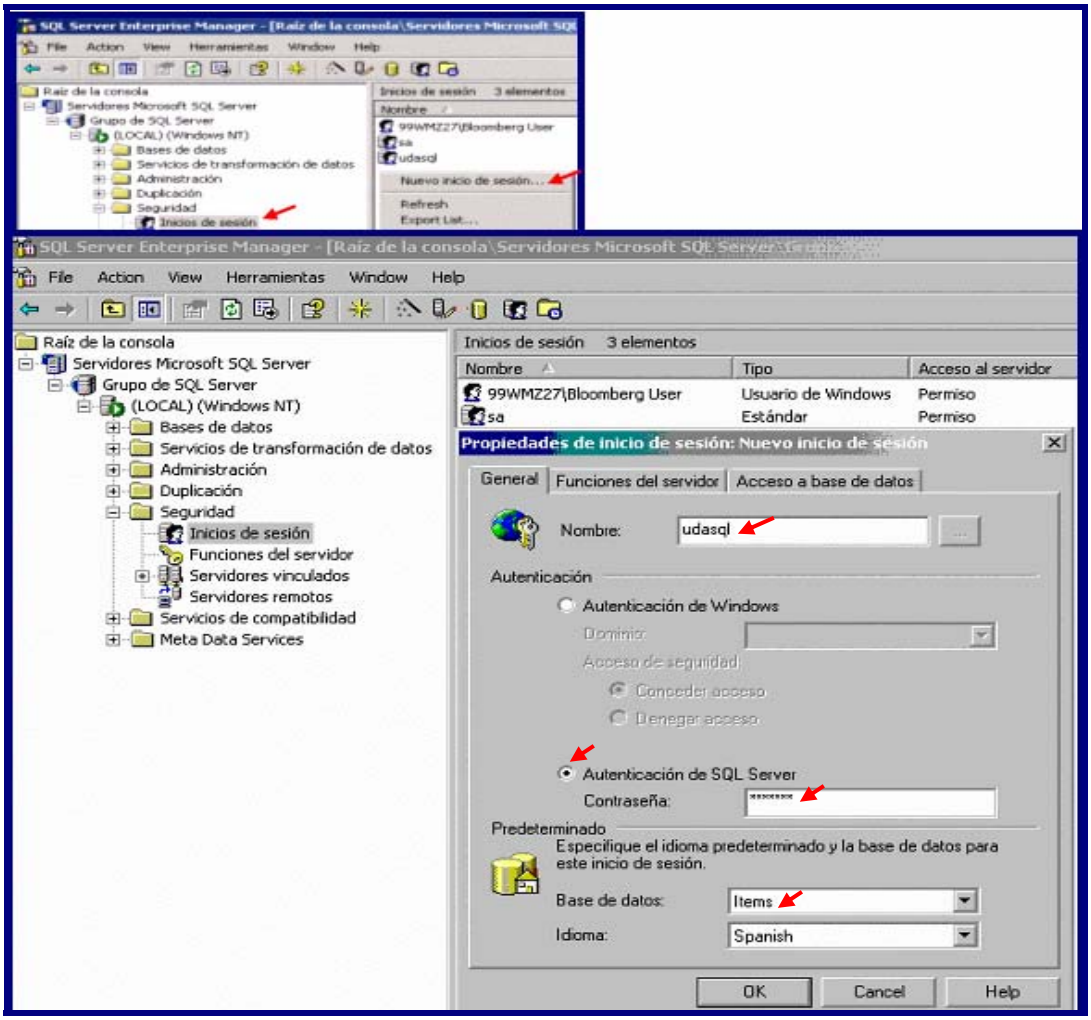


Figura 3.7: Creación de un inicio de sesión y usuario SQL Server con acceso a la base de datos ítems por medio de interfaz gráfica (Administrador Corporativo).

Para acceder a cualquier instancia del gestor, con el inicio de sesión creado, los usuarios deben ingresar tanto el nombre del inicio de sesión como la contraseña asociada.

3.3 Borrado de un inicio de sesión y usuario

Dado que se tiene dos tipos de cuentas de seguridad se deben eliminar ambas para desechar completamente a un usuario del servidor de base de datos, sí se elimina un inicio de sesión se debe eliminar todas las cuentas de usuario asociadas a él, no se puede eliminar un inicio de sesión en uso y nunca se podrá eliminar el inicio de sesión sa. Antes de eliminar un inicio de sesión sería de vital importancia verificar las cuantas de usuario asociadas a él, esto se lo puede hacer mediante funciones del gestor de base de datos (que se los describe al final del capítulo). Para eliminar un inicio de sesión y sus cuentas de usuario se debe ser miembro de los roles de servidor sysadmin o securityadmin, y ejecutar los siguientes procedimientos:

PROCEDIMIENTO	***DESCRIPCION***
sp_revokedbaccess	Procedimiento almacenado del sistema, que elimina una cuenta de usuario sobre un inicio de sesión para cada una de las bases de datos asociadas. Se pasa el nombre de la cuenta de usuario como argumento al procedimiento.
sp_droplogin	Procedimiento almacenado del sistema, requiere únicamente un argumento que especifique el nombre de inicio de sesión a eliminar.

Tabla 3.2: Procedimientos almacenados del sistema.

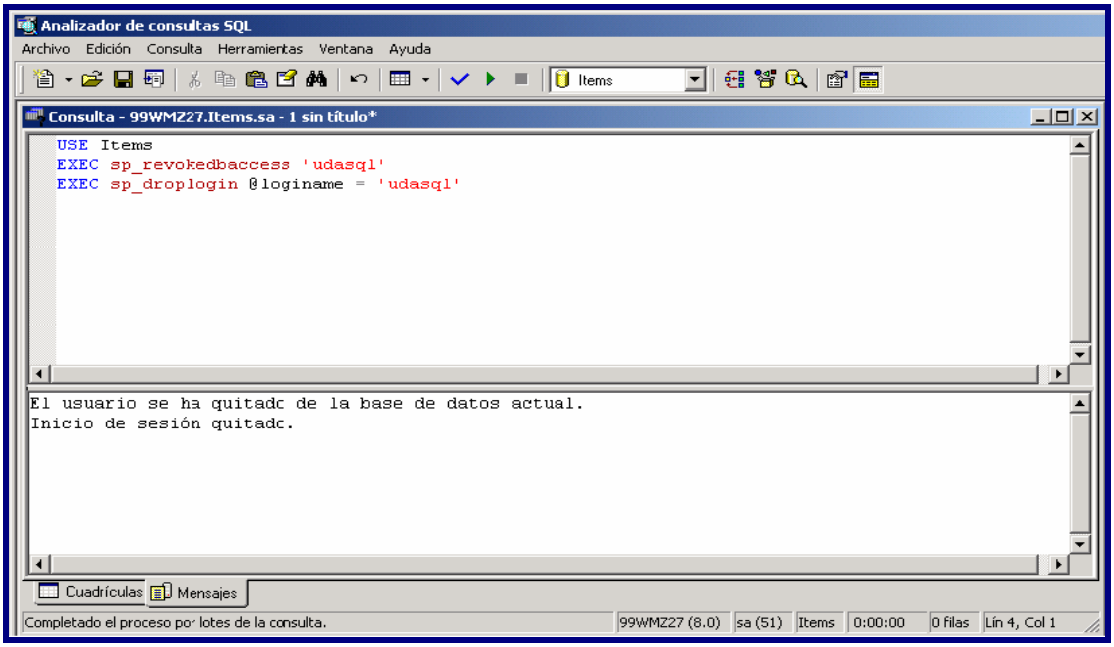


Figura 3.8: Eliminación de un inicio de sesión SQL Server, eliminando primero sus cuentas de usuario.

Por medio del Administrador Corporativo (Interfaz Gráfica), se puede eliminar un inicio de sesión y usuario, ubicándose en el inicio de sesión y dando clic en eliminar, este será borrado automáticamente.

3.4 Creación de inicios de sesión para un usuario Windows

Una cuenta de usuario Windows se entiende como la cuenta por la que Windows valida al usuario, y esta no necesita especificar el inicio de sesión y contraseña para tener acceso a alguna instancia del gestor. Un miembro del grupo sysadmin debe crear un inicio de sesión para la cuenta Windows para que el proceso se realice correctamente. El usuario para ingresar por medio de un inicio de sesión Windows debe seleccionar la opción autenticación Windows en el cuadro de conexión del analizador de consulta. Al momento de crear inicios de sesión para una cuenta de usuario Windows, se llaman a los siguientes procedimientos almacenados del sistema:

PROCEDIMIENTO	***DESCRIPCION***
sp_grantlogin	Crea un inicio de sesión para un usuario Windows, ejemplo: '99WMZ27\Bloomberg User' la parte anterior a la barra es el nombre de servidor Windows, luego de la barra va el nombre de usuario Windows.
sp_grantdbaccess	Crea una cuenta de usuario para un inicio de sesión basado en una cuenta de usuario Windows, pasando como argumento el nombre de inicio de sesión.

Tabla 3.3: Procedimientos almacenados del sistema.

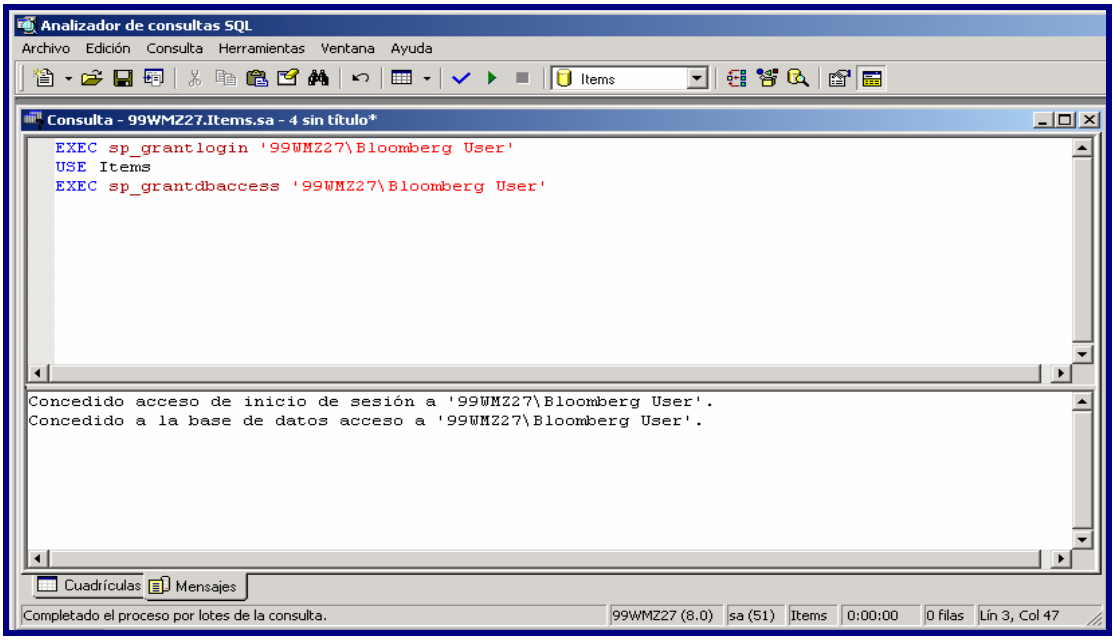


Figura 3.9: Creación de un inicio de sesión Windows con acceso a la base de datos ítems mediante el analizador de consulta.

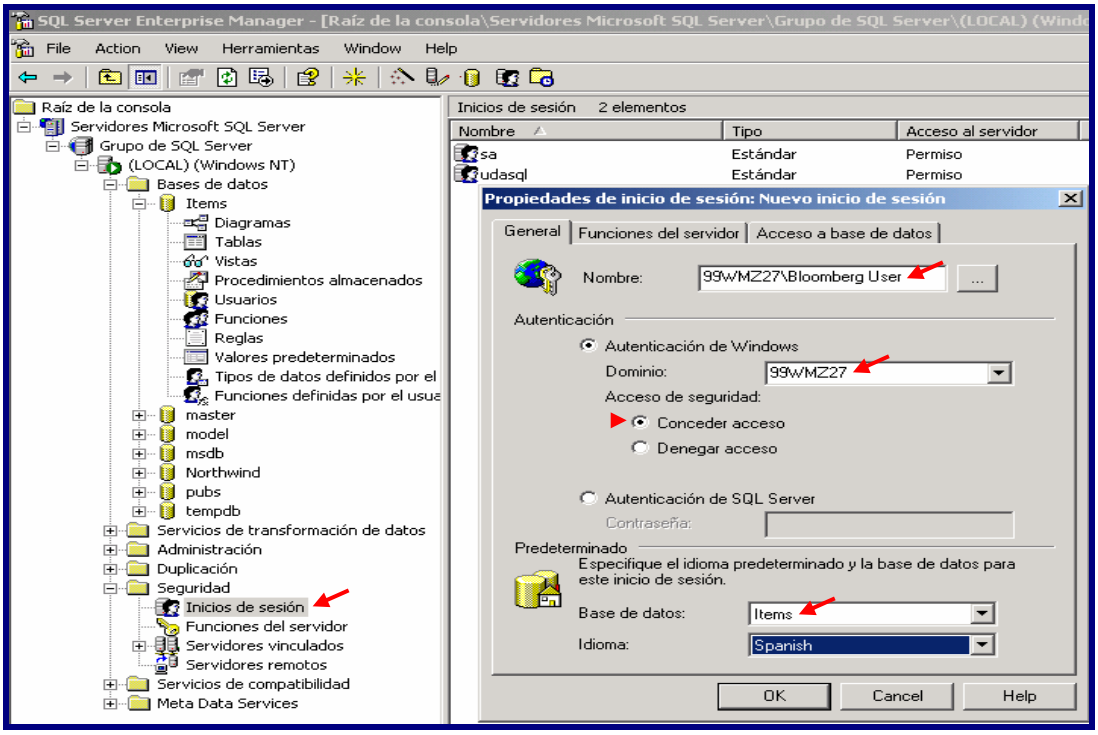


Figura 3.10: Creación de un inicio de sesión Windows, con acceso a la base de datos ítems, mediante interfaz gráfica (Administrador Corporativo).

3.5 Eliminación de inicios de sesión para un usuario Windows

Para eliminar un inicio de sesión y sus cuentas de usuario se debe ser miembro de los roles de servidor sysadmin o securityadmin, y ejecutar los siguientes procedimientos:

PROCEDIMIENTO	***DESCRIPCION***
sp_revokedbaccess	Procedimiento almacenado del sistema, que elimina una cuenta de usuario.
sp_revokelogin	Procedimiento almacenado del sistema, elimina un inicio de sesión Windows.

Tabla 3.4: Procedimientos almacenados del sistema

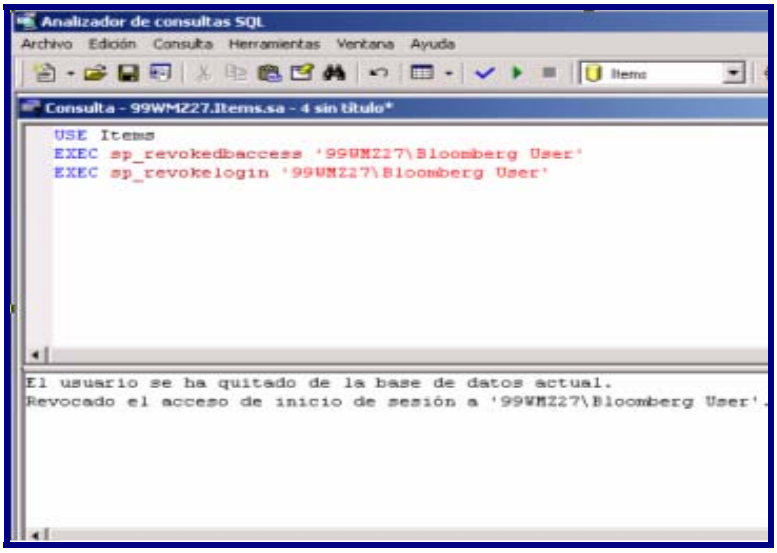


Figura 3.11: Eliminación de un inicio de sesión Windows, eliminando primero la cuenta de usuario.

Con el Administrador Corporativo (Interfaz Gráfica), se puede eliminar un inicio de sesión y usuario Windows, ubicándose en el inicio de sesión y dando clic en eliminar, este será borrado automáticamente.

3.6 Funciones del gestor de base de datos

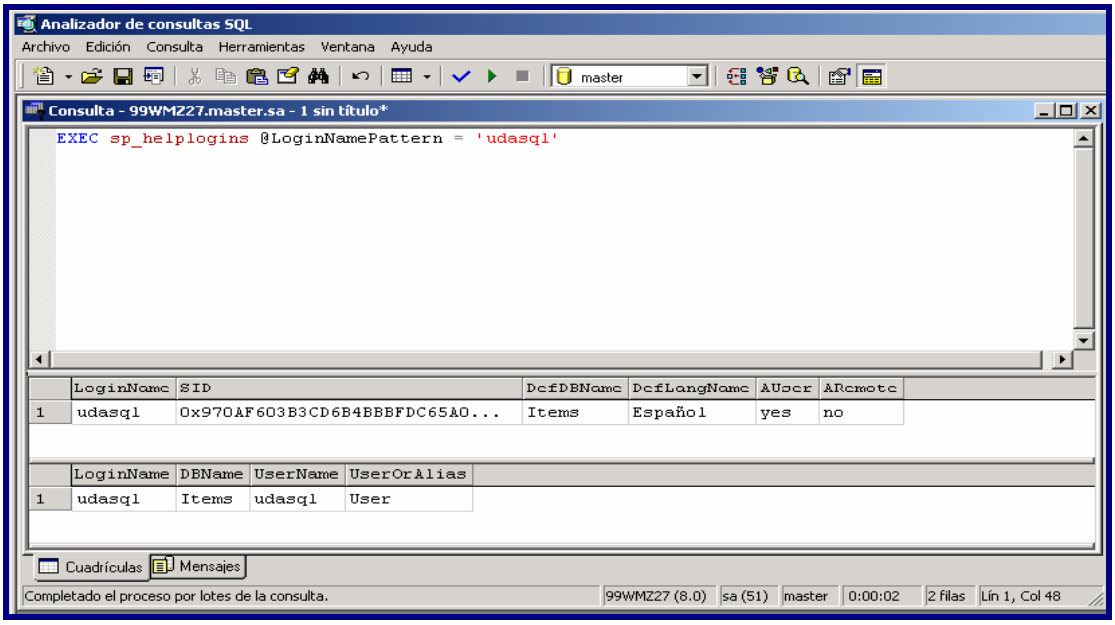


Figura 3.12: Obtiene información sobre un inicio de sesión, incluyendo las cuentas de usuario de base de datos.

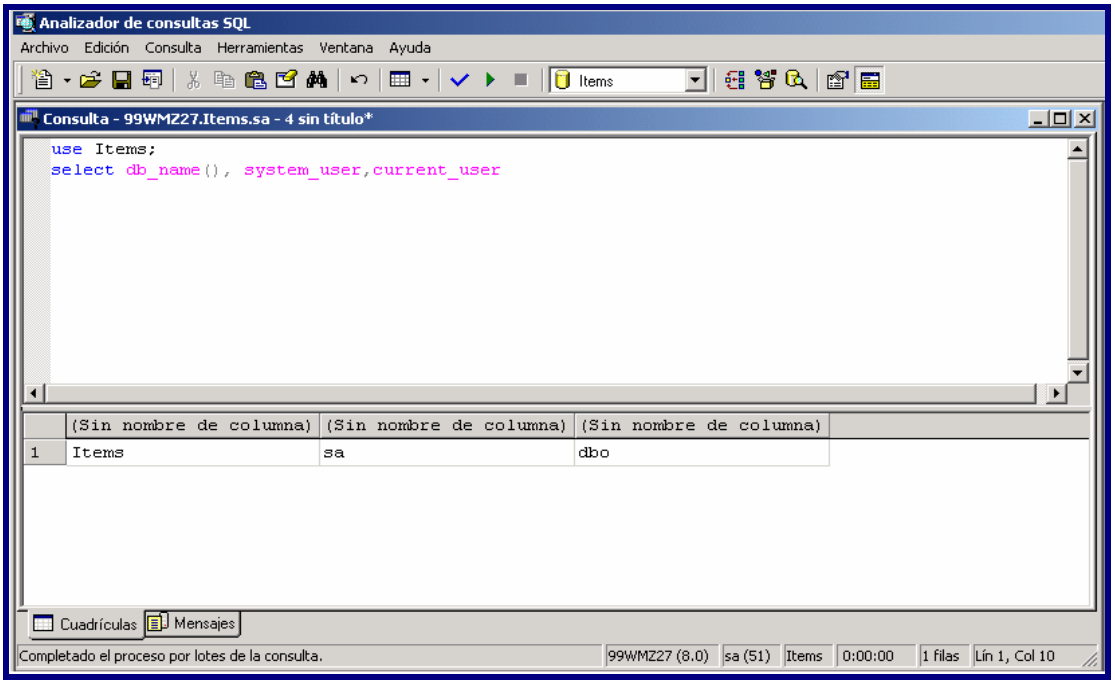


Figura 3.13: Demuestra las funciones que indican el nombre de inicio de sesión y la cuenta de usuario de una base de datos.

3.7 Conclusiones

Las creación y borrado de un inicio de sesión y usuario es importante para el manejo adecuado de las bases de datos, ya que mediante esto los usuarios pueden realizar funciones de servidor y base de datos que son necesarias en su labor; inclusive administrar su seguridad, además mediante la creación y borrado de un inicio de sesión y usuario se puede administrar la confidencialidad, que se cumple cuando únicamente las personas autorizadas pueden conocer y tener acceso a una determinada información.

CAPITULO 4: INSTRUCCIONES DE ASIGNACION, REVOCACION Y NEGACION DE PERMISOS

Introducción

En este capítulo se pretende conocer acerca de los permisos de concesión, revocación y negación hacia y desde los usuarios de base de datos, considerando los permisos para roles definidos por el usuario y roles de aplicación. Recuerde que un permiso obtenido por medio de la sentencia GRANT sobrescribe una sentencia REVOKE y una sentencia DENY sobrescribe cualquier sentencia GRANT.

4.1 Instrucción GRANT

Concede los permisos específicos (SELECT, INSERT, DELETE Y UPDATE) y las instrucciones (CREATE DATABASE, CREATE DEFAULT, CREATE FUNCTION, CREATE PROCEDURE, CREATE RULE, CREATE TABLE, CREATE VIEW, BACKUP DATABASE Y BACKUP LOG) a un usuario, a un rol definido por el usuario o a un rol de aplicación. La instrucción GRANT, dependiendo de los permisos e instrucciones que se concedan y del objeto implicado en este, pueden ser ejecutados por miembros de la función sysadmin, por los propietarios de objetos o por los miembros de las funciones db_owner o db_securityadmin.

Permisos de la instrucción:

```
GRANT { ALL | statement [ ,...n ] }
TO security_account [ ,...n ]
```

Permisos del objeto:

```
GRANT
   { ALL  | permission [ ,...n ] }
   {
      [ ( column [ ,...n ] ) ] ON { table | view }
      | ON { table | view } [ ( column [ ,...n ] ) ]
      | ON { user_defined_function }
   }
TO security_account [ ,...n ]
[ WITH GRANT OPTION ]
[ AS { group | role } ]
```

ARGUMENTO	***DESCRIPCION***
ALL	Otorga todos lo permisos que se pueden aplicar.
statement	Son las instrucciones (descritos anteriormente) para la que se concede el permiso.
N	Indica que el elemento se puede repetir en una lista separada por comas.
TO	Determina las cuentas de seguridad.
Security_account	Cuenta de seguridad (Usuario y Función SQL o Usuario de Windows) a la que se aplican los permisos. La cuenta especificada debe existir en la base de datos actual.
permission	Son permisos de objeto que se conceden tanto a usuarios como a roles (SELECT, INSERT, DELETE o UPDATE). Se puede dar permisos a una lista de columnas junto con los permisos SELECT y UPDATE de lo contrario los permisos se aplican a todas las columnas de la tabla o vista.
column	Nombre de una columna de una tabla de la base de datos actual sobre la que se conceden los permisos.
Table	Nombre de la tabla de la base de datos actual a la que se desea conceder permisos.
View	Nombre de la vista de la base de datos actual a la que se conceden permisos.
user_defined_function	Nombre de la función definida por el usuario a la que se conceden permisos.
WITH GRANT OPTION	WITH GRANT OPTION concede a *security_account* la capacidad de conceder el permiso del objeto especificado a otras cuentas de seguridad, está es válida únicamente con los permisos de objeto.
AS	Se utiliza cuando se conceden permisos sobre un objeto a un rol definido por el usuario o a un rol de aplicación, y es necesario que los permisos de objetos se concedan además a otros usuarios que no son miembros de los roles.

[1] Tabla 4.1: Argumentos de la sentencia GRANT.

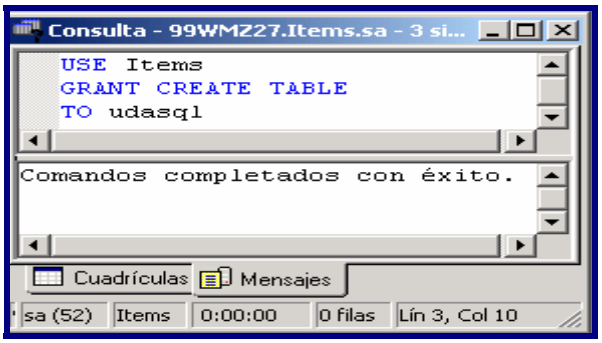


Figura 4.1: Concede permisos sobre instrucciones a usuarios Sql o Windows, definiendo únicamente el nombre de usuario.

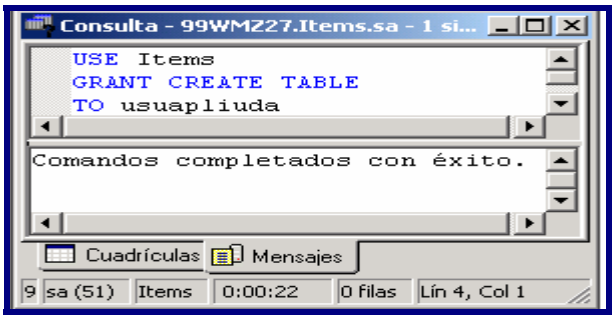


Figura 4.2: Concede permisos sobre instrucciones a funciones definidas por el usuario y funciones de aplicación.

[1] Libros en pantalla de SQL Server. Versión 2000.

Con la siguiente descripción se puede conceder permisos con las opciones WITH GRAN OPTION y la cláusula AS. Udasql concede permisos sobre la función usuapliuda. El usuario usuprueba, miembro de usuapliuda, concede permisos sobre la tabla artículos al usuario prueba, que no es miembro de usuapliuda. Usuprueba no puede conceder permisos sobre la tabla basándose en los permisos que ha recibido por ser miembro de la función usuapliuda. Usuprueba debe utilizar la cláusula AS para conceder permisos pertenecientes a la función usuapliuda.

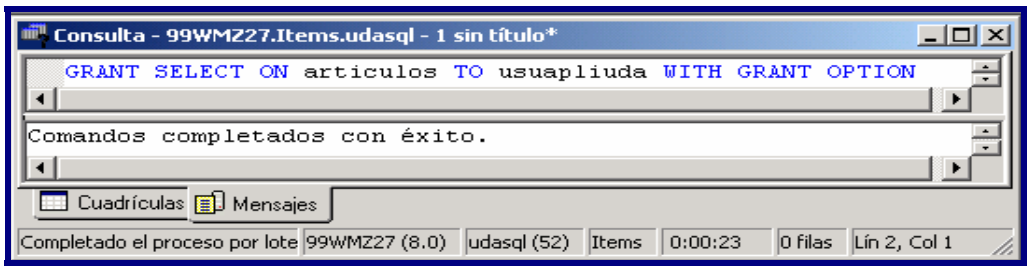


Figura 4.3: Concesión de permisos a la función usuapliuda.

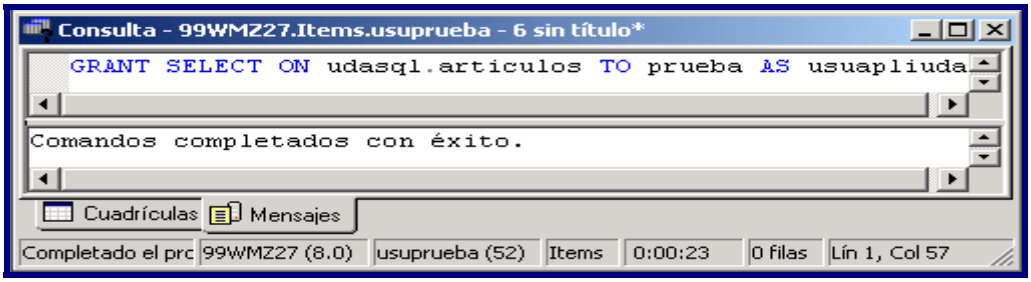


Figura 4.4: Concesión de permisos a un usuario que no es miembro de una función.

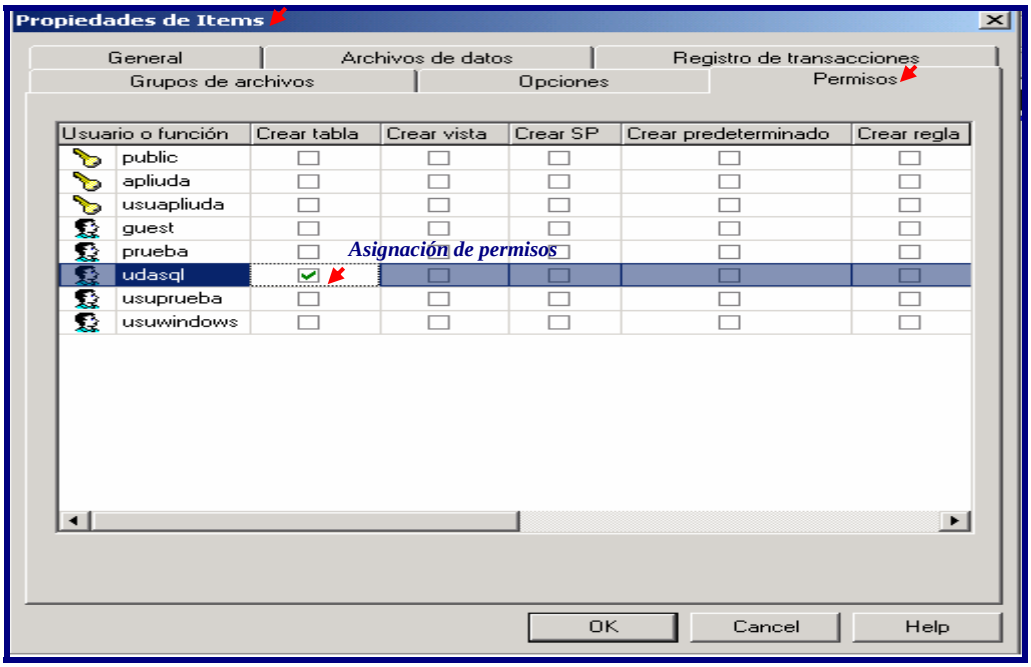


Figura 4.5: Concede permisos sobre instrucciones a un usuario, o a una función, mediante interfaz gráfica.

4.2 Instrucción REVOKE

Remueve un permiso, ya sea GRANT o DENY de un usuario o rol en la base de datos actual. Los miembros de sysadmin, db_owner, db_securityadmin, y los propietarios de objetos de base de datos pueden ejecutar la sentencia REVOKE.

Figura 4.6: Remueve permisos sobre instrucciones a usuarios Sql o Windows, definiendo únicamente el nombre de usuario.

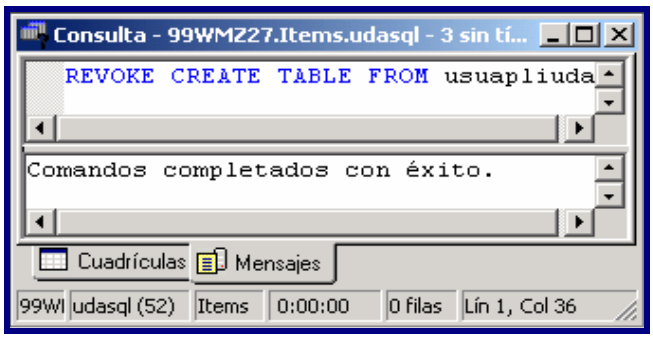


Figura 4.7: Remueve permisos sobre instrucciones a funciones definidas por el usuario y funciones de aplicación.

Mediante la siguiente aplicación se demuestra la revocación de los permisos WITH GRANT OPTION. La cuenta de seguridad permanece con los permisos otorgados pero no puede otorgarlos a otros usuarios. Se debe especificar las cláusulas CASCADE y GRANT OPTION FOR cuando el permiso es otorgado a otro usuario por parte de la cuenta de seguridad.

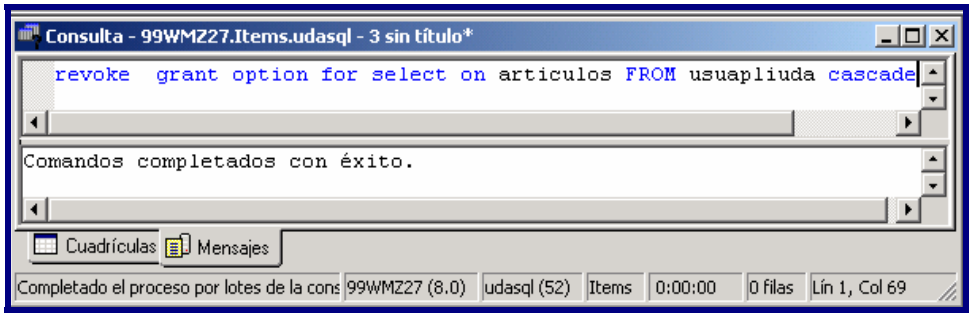


Figura 4.8: Remueve la concesión de permisos de una cuenta de seguridad y los permisos otorgados a otros usuarios mediante esta cuanta.

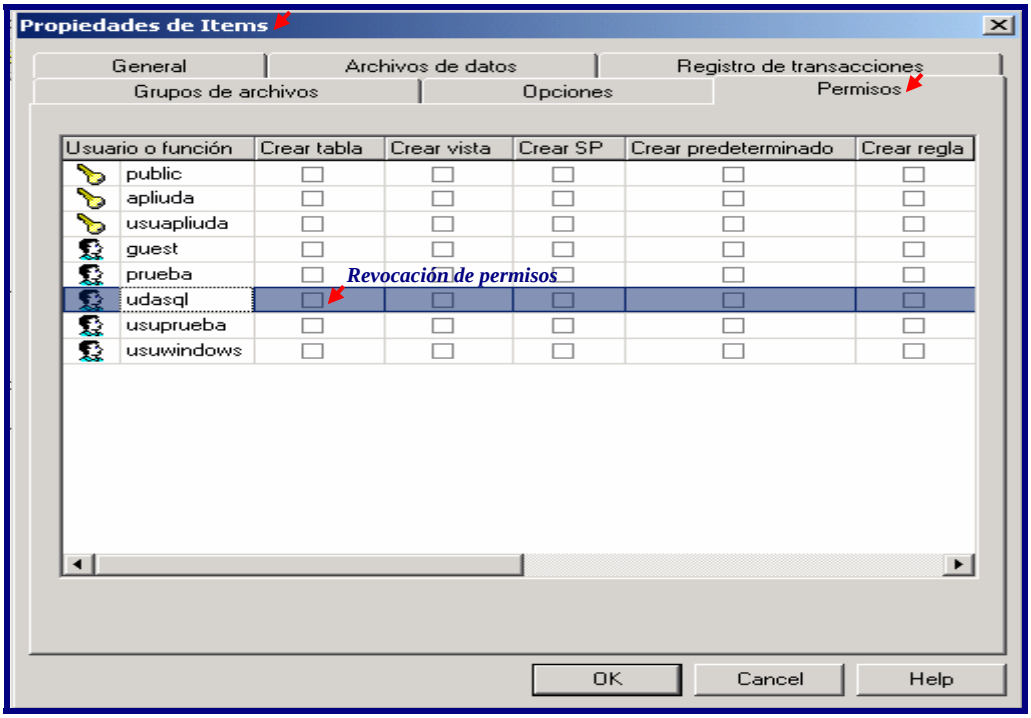


Figura 4.9: Remueve permisos sobre instrucciones a un usuario o a una función, mediante interfaz gráfica.

4.3 Instrucción DENY

Niega un permiso a un usuario o rol de la base de datos actual e impide que la cuenta de seguridad herede permisos a través de funciones. Los permisos para utilizar DENY pertenecen a los miembros de las funciones sysadmin, db_owner y db_securityadmin, y a los propietarios de objetos de bases de datos.

Figura 4.10: Niega permisos sobre instrucciones a usuarios Sql o Windows, definiendo únicamente el nombre de usuario.

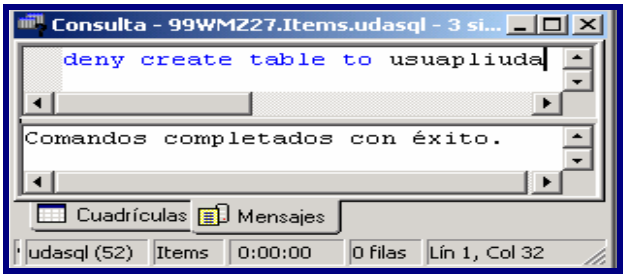


Figura 4.11: Niega permisos sobre instrucciones a roles definidos por usuario o de aplicación.

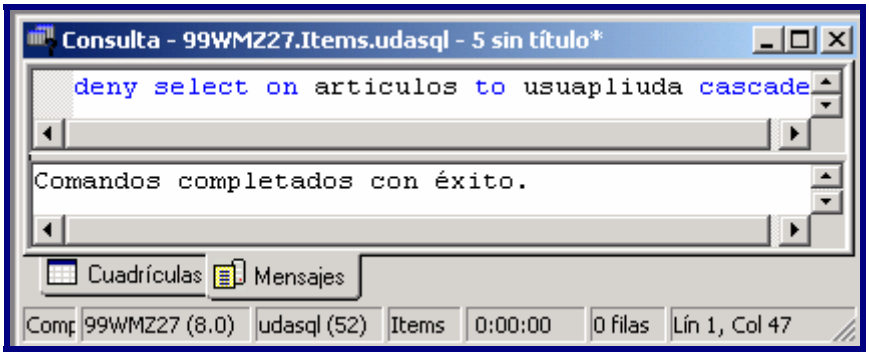


Figura 4.12: Niega permisos a una cuenta de seguridad y los permisos concedidos a los usuarios por parte de la cuenta de seguridad.

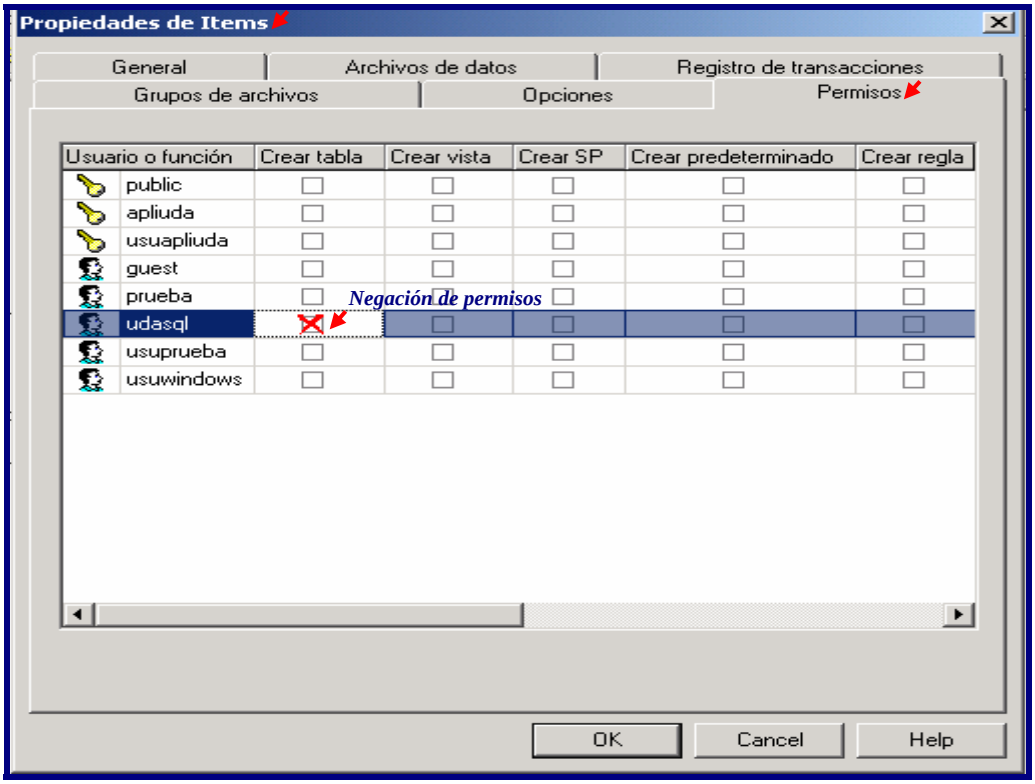


Figura 4.13: Niega permisos sobre instrucciones a un usuario, o a una función definida, mediante interfaz gráfica.

4.4 Conclusiones

Fundamental es conocer los privilegios de concesión, revocación y negación de permisos tanto para usuarios como para funciones, ya que mediante esto se puede lograr tener una convicción de que los datos alcancen una protección adecuada y que la información sea accedida por personas que tienen autorización y obligaciones sobre la misma. Una vez culminado el estudio de estos comandos podemos darnos cuenta de que se puede implementar una seguridad efectiva, sí los mismos son empleados de una manera eficiente.

CAPITULO 5: ASIGNAR PERMISOS Y HABILITAR AUDITORIA

Introducción

Esta sección demuestra las diferentes sentencias para organizar los permisos dentro de una base de datos determinada, administrando los mismos para una cuenta de usuario, una vista, un rol definido por el usuario, o un rol de aplicación. Además en este capítulo se analiza el nivel de auditoría con el que cuenta el servidor de base de datos.

5.1 Asignación de permisos SELECT, INSERT, DELETE y UPDATE

Instrucción SELECT

- Al asignar el permiso Select se especifica que una o varias columnas de una tabla pueden ser consultadas o leídas en la base de datos actual.

Instrucción INSERT

- Al conceder el permiso de inserción a un usuario o a una función, se agrega una nueva fila a una tabla o vista en la base de datos determinada.

Instrucción DELETE

- Por medio de la asignación de este permiso se quitan filas de una tabla en una determinada base de datos.

Instrucción UPDATE

- Con la sentencia de modificación se dan permisos a un usuario o función para cambiar datos de una tabla en la base de datos actual.

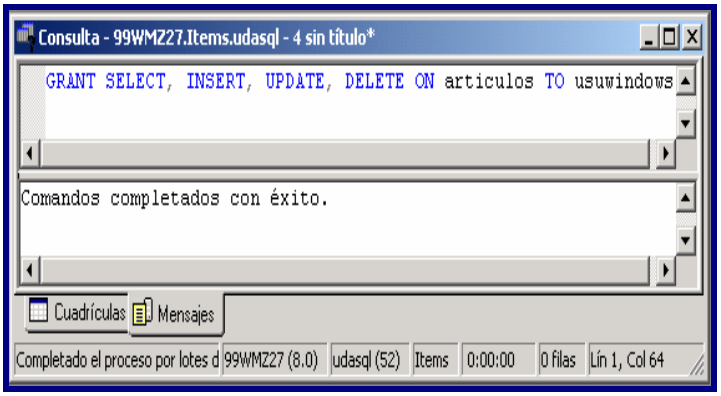


Figura 5.1: Asignación de permisos de objeto a usuarios Sql o Windows.

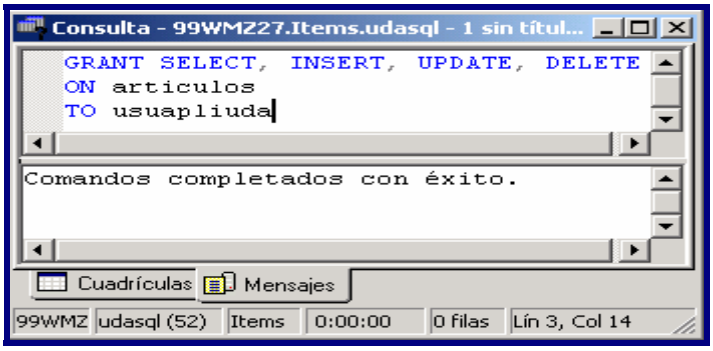


Figura 5.2: Asignación de permisos de objeto a funciones.

Mediante las vistas se puede implementar el permiso SELECT en una fila concreta a un usuario uno y denegar el permiso SELECT en otra fila a un usuario dos. Pero esta solución puede llegar a ser un poco tediosa en el caso de tener muchos usuarios con requerimientos de datos variados.

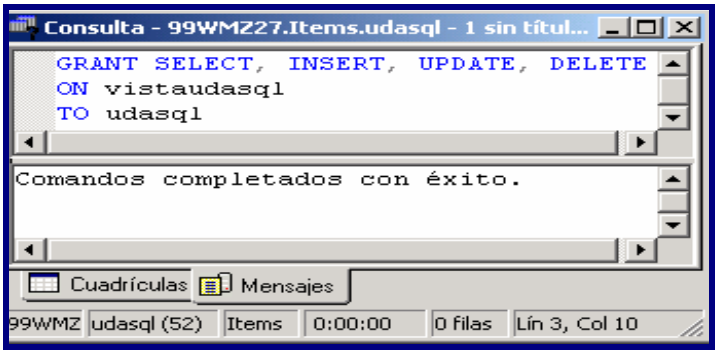


Figura 5.3: Asignación de permisos de objeto mediante vistas, para un usurio.

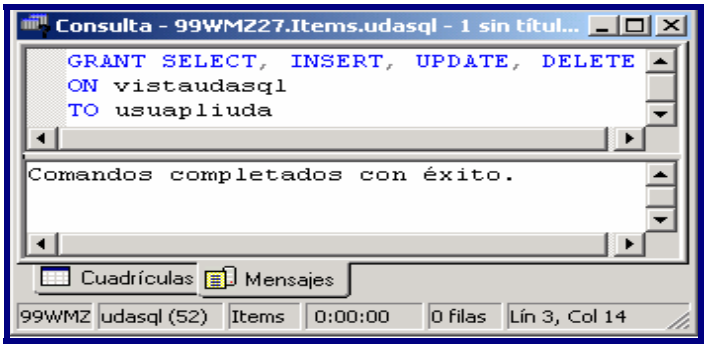


Figura 5.4: Asignación de permisos de objeto para una función mediante vistas.

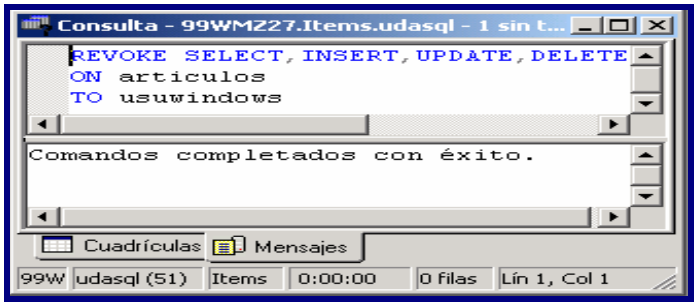


Figura 5.5: Revocación de permisos de objeto a usuarios Sql o Windows.

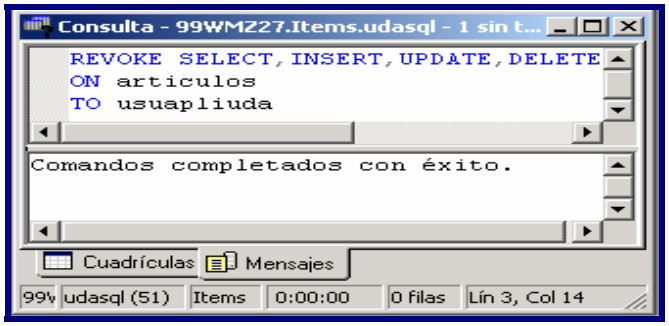


Figura 5.6: Revocación de permisos de objeto a funciones.

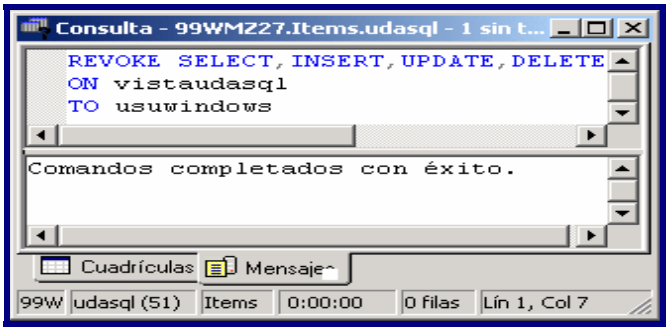


Figura 5.7: Revocación de permisos de objeto sobre vistas.

Figura 5.8: Negación de permisos de objeto a usuarios Sql o Windows mediante la sentencia ALL.

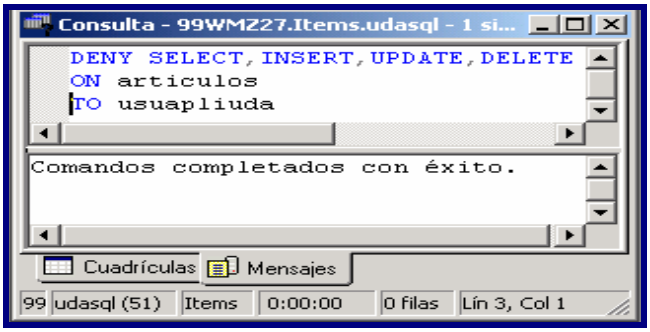


Figura 5.9: Negación de permisos de objeto a funciones.

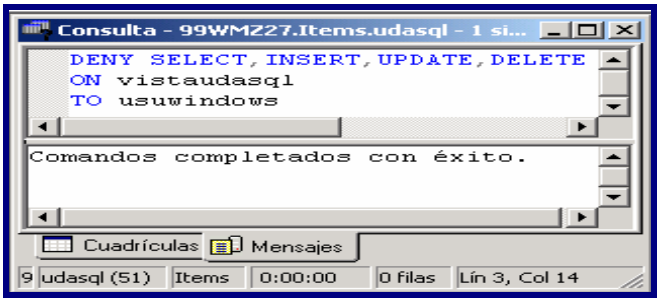


Figura 5.10: Negación de permisos de objeto sobre vistas.

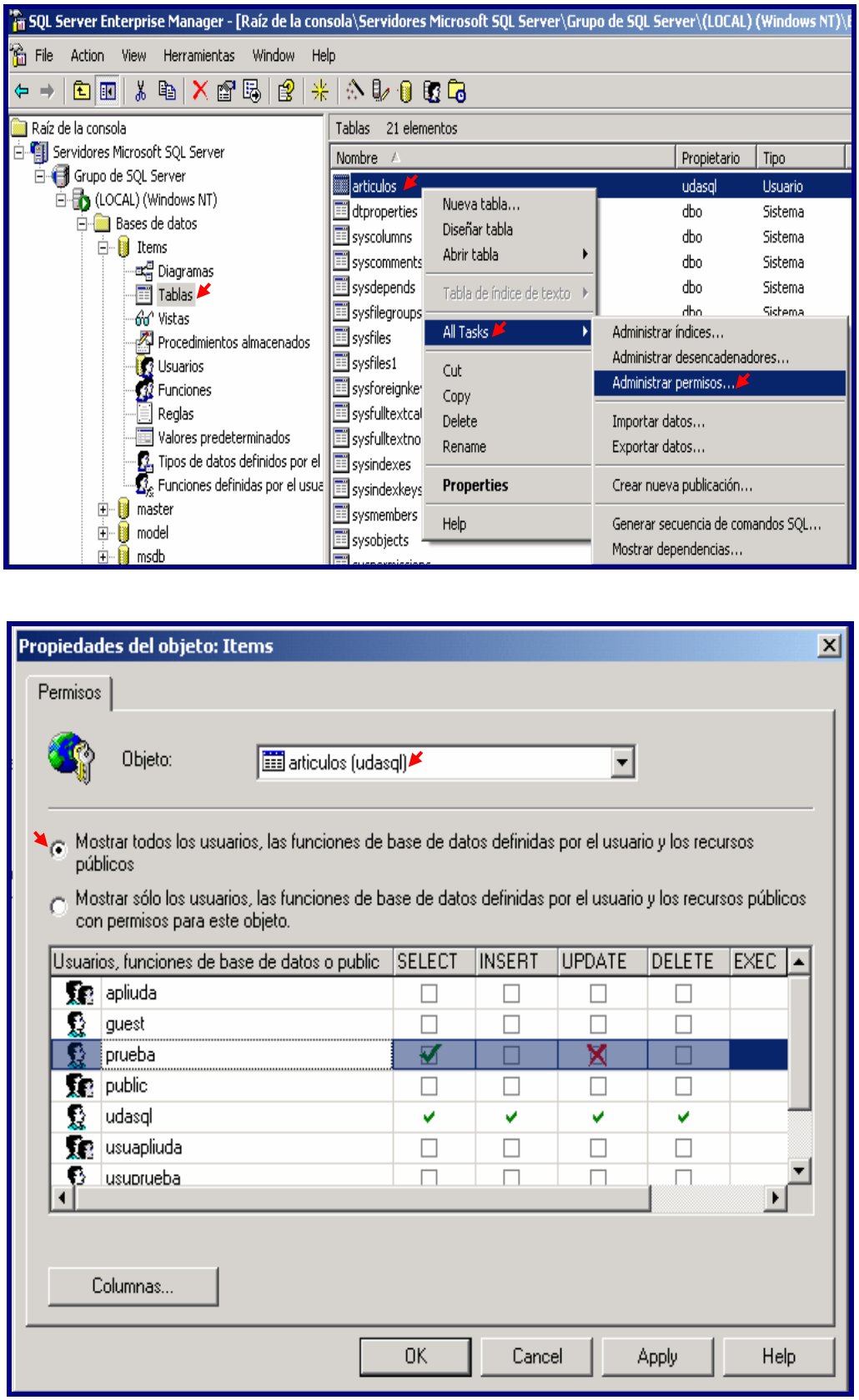


Figura 5.11: Asignación, revocación y negación de permisos de objeto sobre una tabla de la base de datos, por medio de interfaz gráfica.

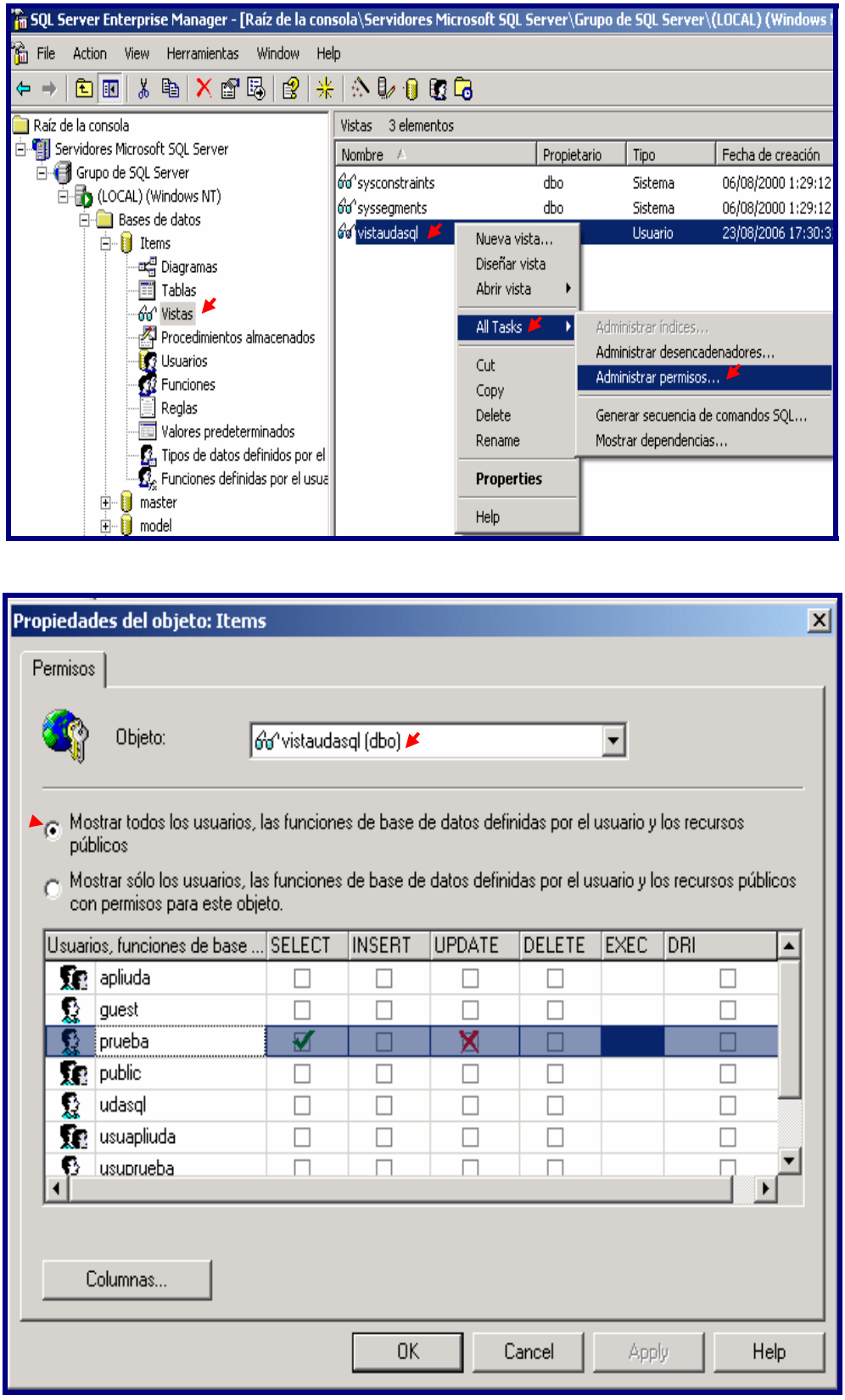


Figura 5.12: Asignación, revocación y negación de permisos de objeto sobre una vista de la base de datos, por medio de interfaz gráfica.

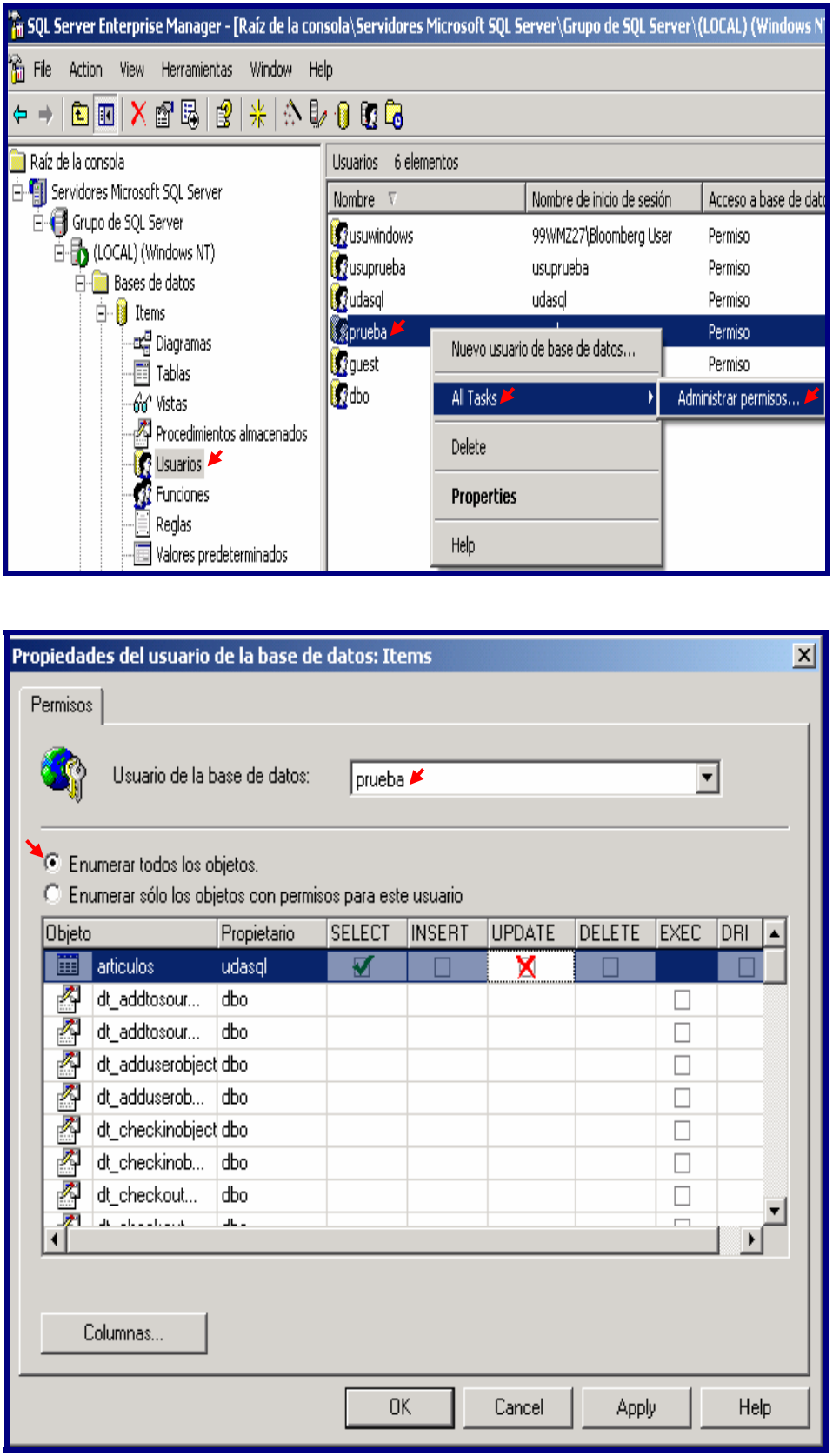
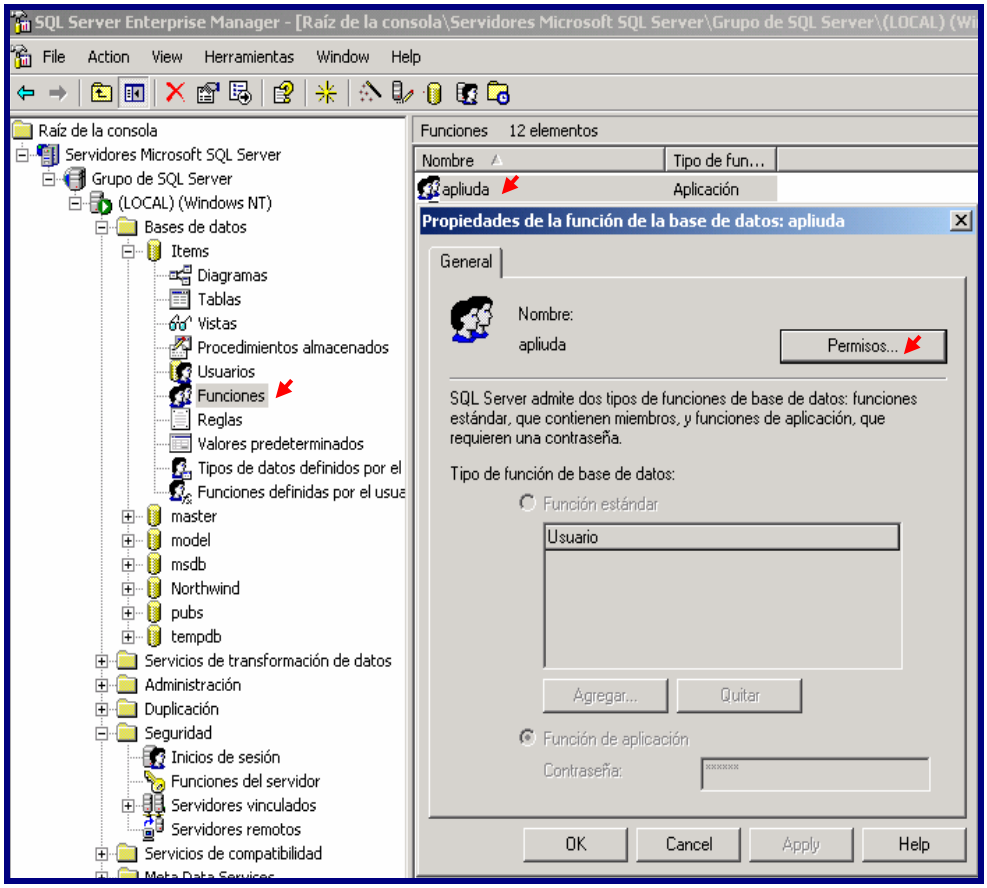


Figura 5.13: Asignación, revocación y negación de permisos de objeto sobre un usuario de la base de datos, por medio de interfaz gráfica.



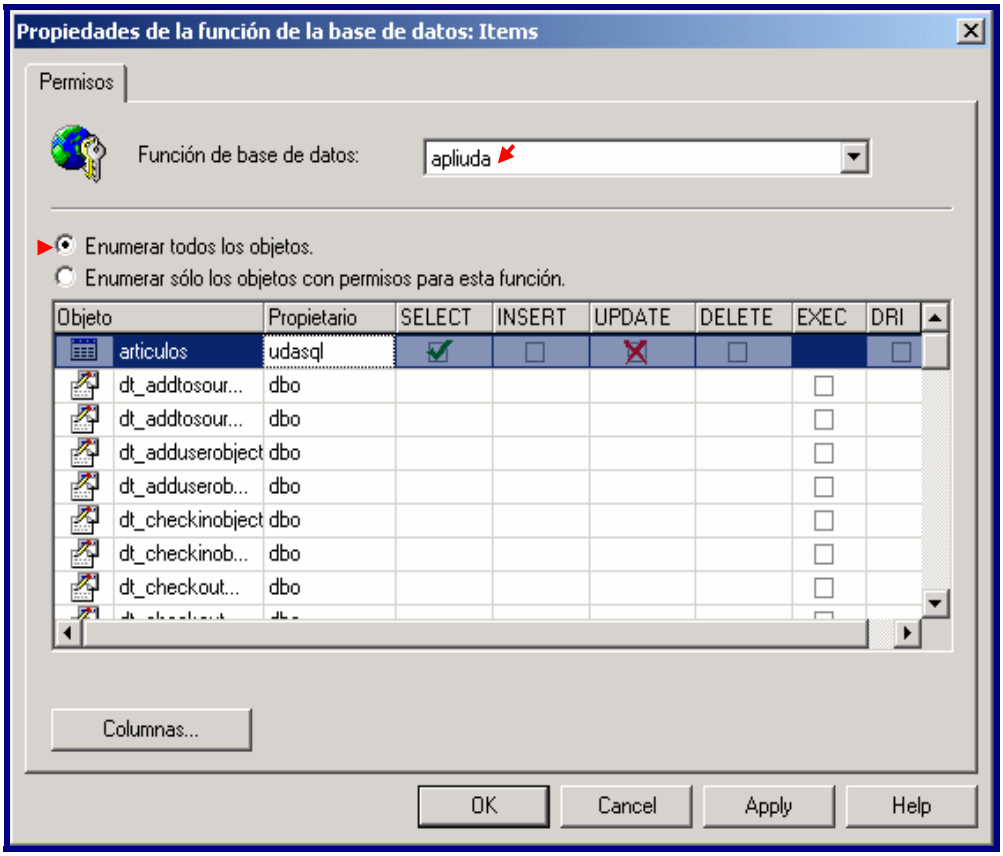


Figura 5.14: Asignación, revocación y negación de permisos de objeto sobre una función de la base de datos, por medio de interfaz gráfica.

5.2 Asignación de permisos SELECT y UPDATE sobre columnas de una tabla

El gestor de base de datos únicamente permite trabajar con permisos Select y Update sobre columnas. Para tener acceso a una columna en las instrucciones antes mencionadas es necesario disponer de permisos en esa columna para ejecutar dichas sentencias.

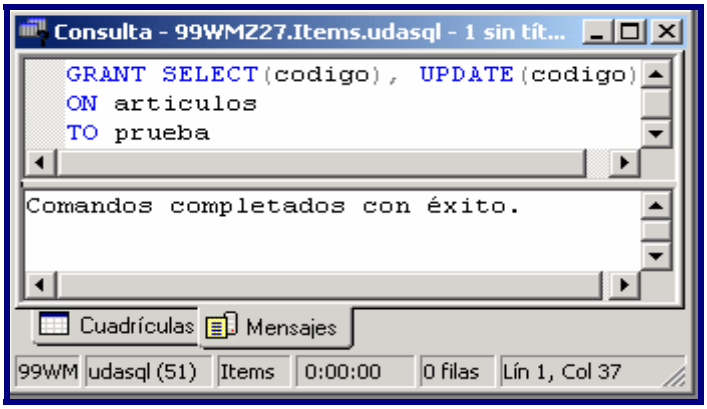


Figura 5.15: Asignación de permisos de objeto a usuarios Sql o Windows.

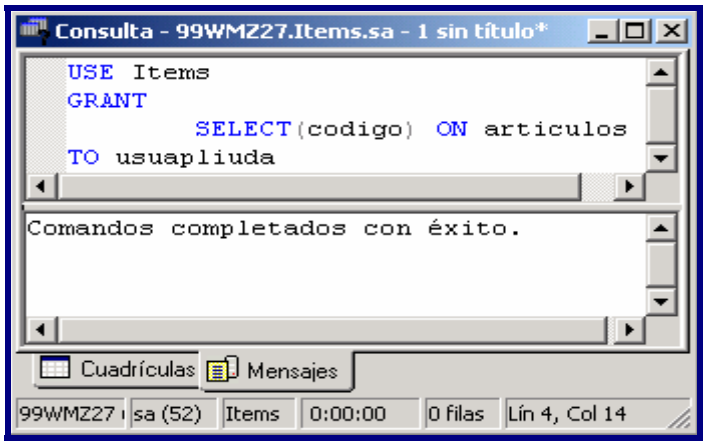


Figura 5.16: Asignación de permisos de objeto a funciones.

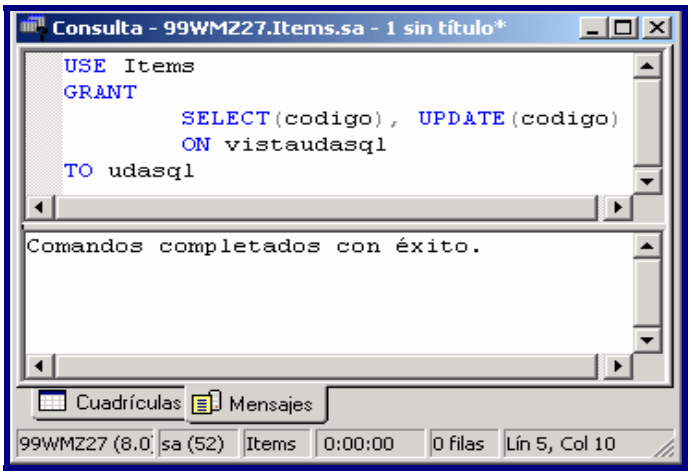


Figura 5.17: Asignación de permisos de objeto sobre una vista.

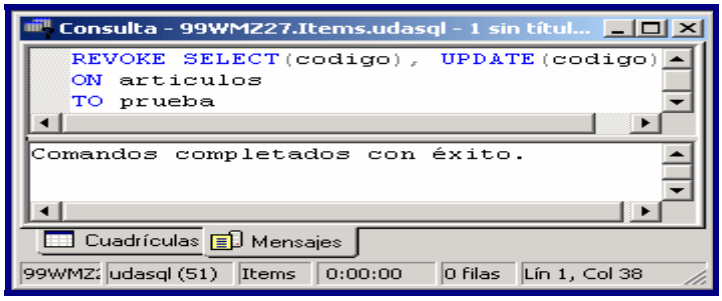


Figura 5.18: Revocación de permisos de objeto.

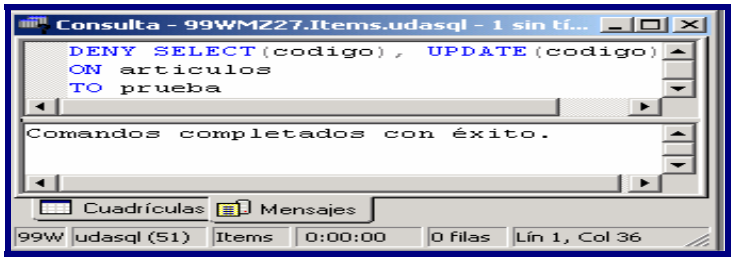


Figura 5.19: Negación de permisos de objeto.

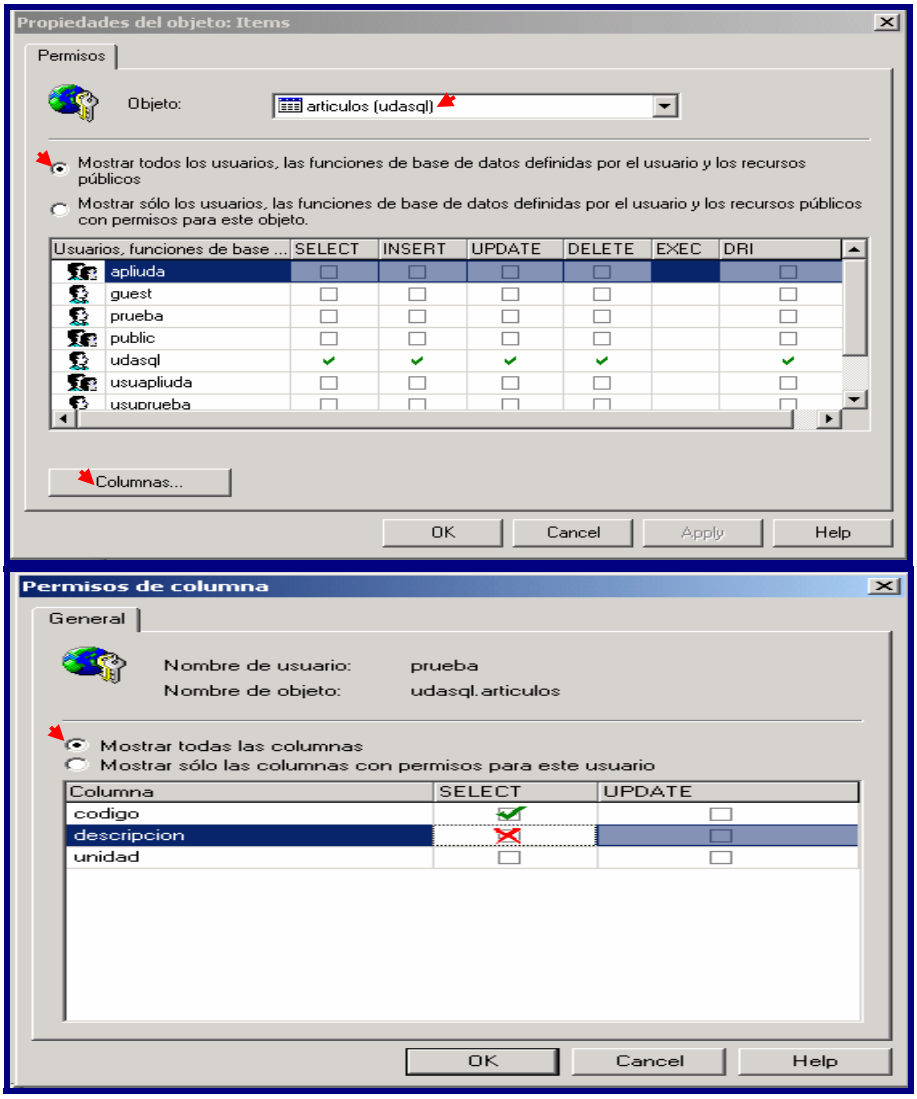


Figura 5.20: Asignación, revocación y negación de permisos de objeto sobre columnas de una tabla, mediante interfaz gráfica.

5.3 Habilitar auditoría

La habilitación de auditoría es importante, cuando se requiere saber que usuarios y cuantas veces han intentado conectarse en forma correcta y errónea al servidor de base de datos. Se recomienda crear un directorio nuevo durante la instalación del gestor de base de datos, para alojar los archivos de auditoría. La ruta de acceso recomendada es \MSSQL\AUDIT. Unicamente los miembros de la función fija de seguridad sysadmin pueden habilitar o modificar la auditoría, y toda modificación es un suceso auditable. [1]

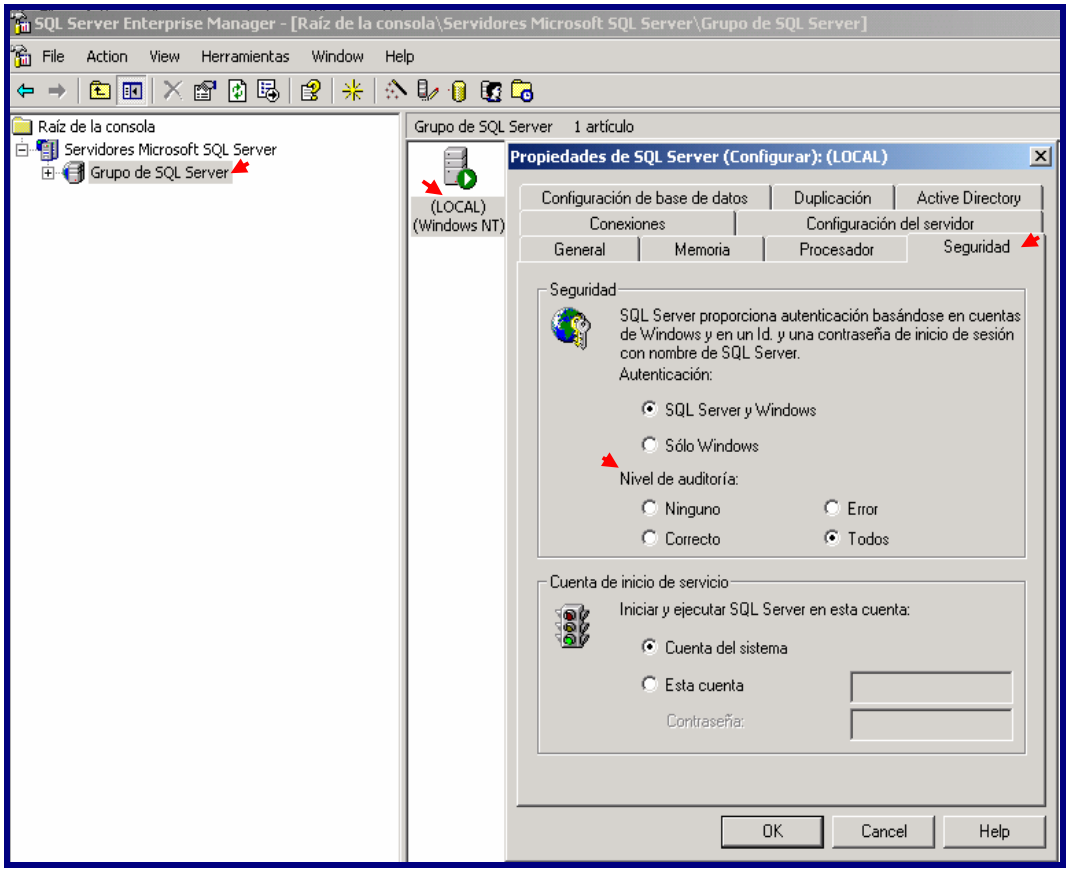


Figura 5.21: Habilitar auditoría.

5.3.1 Nivel de auditoría

- Ninguno, hace que no se realice la auditoría, este es el valor predeterminado para esta configuración.

- Correcto, hace que solo se incluyan en el registro de auditoría los inicios de sesión que tengan éxito.

[1] Libros en pantalla de SQL Server. Versión 2000.

- Error, hace que solo se incluyan en el registro de auditoría los intentos de inicio de sesión erróneos.

- Todos, hace que se incluyan en el registro de auditoría los intentos de inicio de sesión con éxito y los erróneos.

Los niveles de auditoría correcto, erróneo y todos, al seleccionarlos deben detener y reiniciar el servidor para habilitar la auditoría.

5.4 Conclusiones

La asignación de permisos para listar, insertar, modificar y eliminar objetos dentro de una base de datos, permite una protección adecuada para la información frente a amenazas.

La auditoría dentro del servidor de base de datos es fundamental por que al momento de aplicarla se indaga sobre que usuarios y cuantas veces se ha intentado conectar en forma errónea; por medio de este mecanismo se enfrenta amenazas, que pueden ser cruciales para el buen mantenimiento de la información.

CAPITULO 6: OTRAS SEGURIDADES

Introducción

La seguridad se basa en políticas implementadas sobre una base de continuidad para desarrollar medidas que aporten en contra de posibles ataques a la información de una empresa. Esto conlleva a tener especial cuidado en la seguridad de cuentas del servidor y la realización del backup.

6.1 Seguridad en la cuentas

Una de las seguridades fundamentales para el acceso a base de datos es comprobar que no existan cuentas sin contraseñas, especialmente si se trata de cuentas de invitado. Además se debe tener cuidado con las cuentas de administrador, que tienen acceso completo a todas las bases de datos; el gestor de base de datos no permitirá borrar estas cuentas, por lo que se debe asignar una contraseña aleatoria a dichas cuentas.[1]

6.2 Seguridad física del servidor

Todo lugar en donde se encuentre ubicado un servidor deberá tener llave y la misma deberá estar en manos de un responsable, restringiendo el acceso físico únicamente al personal autorizado. Se deberá tener en cuenta la seguridad física dentro del espacio donde está ubicado o dispuesto el servidor. Se debe considerar lo siguiente: [2]

- Extintores, ante incendios o corto circuitos.
- Alarmas y sistemas de estabilización de la tensión.
- Mantener una temperatura constante (refrigeración y calefacción).
- Ubicación adecuada del servidor, por ejemplo si existen riesgos de inundación, se recomienda ubicarlo en un lugar alto.
- El cableado debe estar adecuado al lugar para evitar problemas con el fluido eléctrico.
- UPS o sistema de alimentación continua de energía. [2]

[1] FARLEY. Marc. HSU. Jeffrey. STEARNS. Tom. Seguridad e Integridad de Datos. España: Madrid, 1998. Pág. 175.

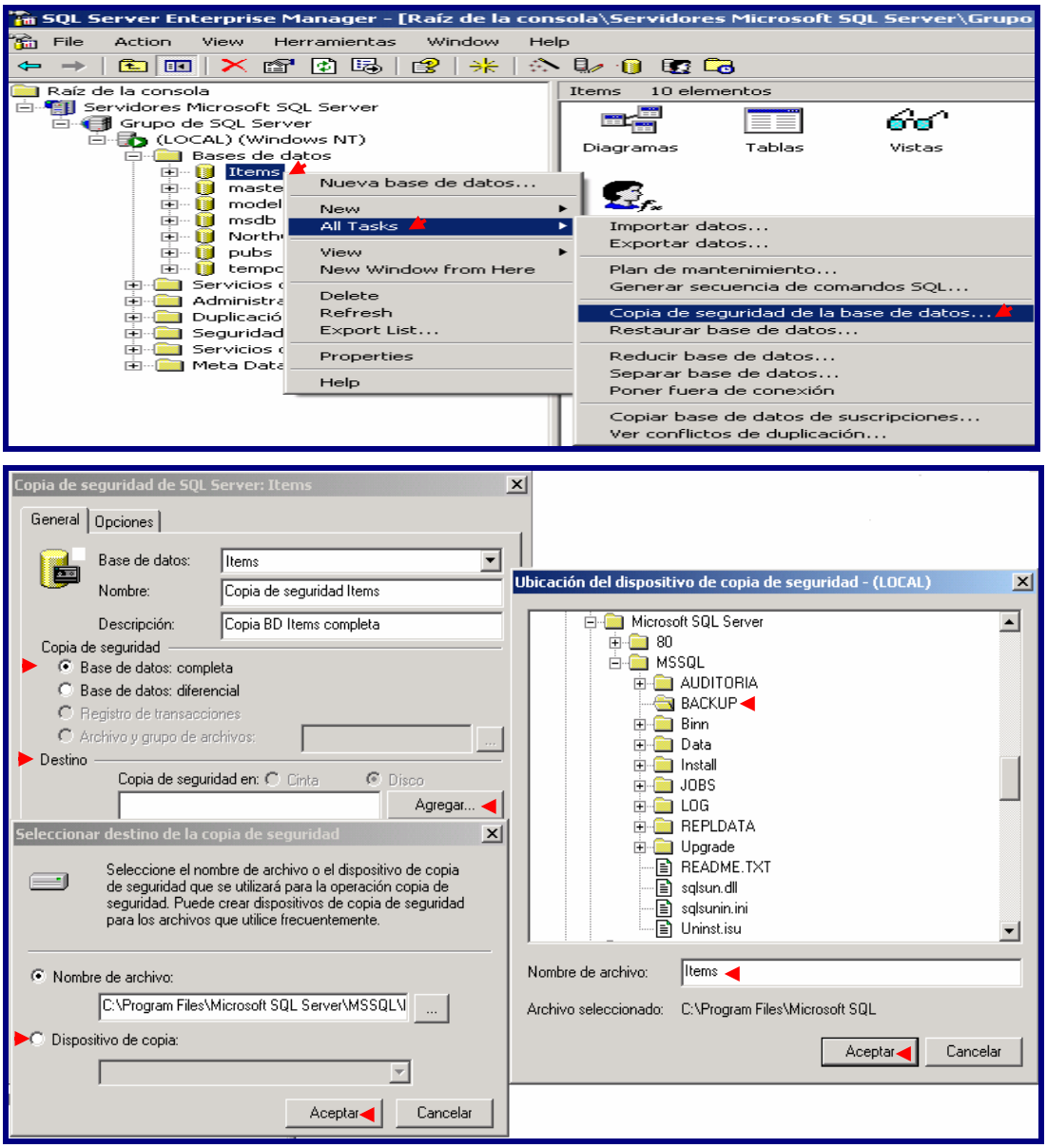
[2] ANCAJIMA Robert. Seguridad en SQL SERVER 2000. http://www.informatizate.net 2004. [consulta 20 de julio de 2006]

6.3 Copias de seguridad

Las copias de seguridad son mecanismos de recuperación que poseen los administradores para rescatar información perdida; es fundamental en la planificación de la seguridad. Los problemas de copias de seguridad se dan cuando no se realizan las mismas. Esto sucede de manera frecuente y el resultado puede ser desastroso. [3]

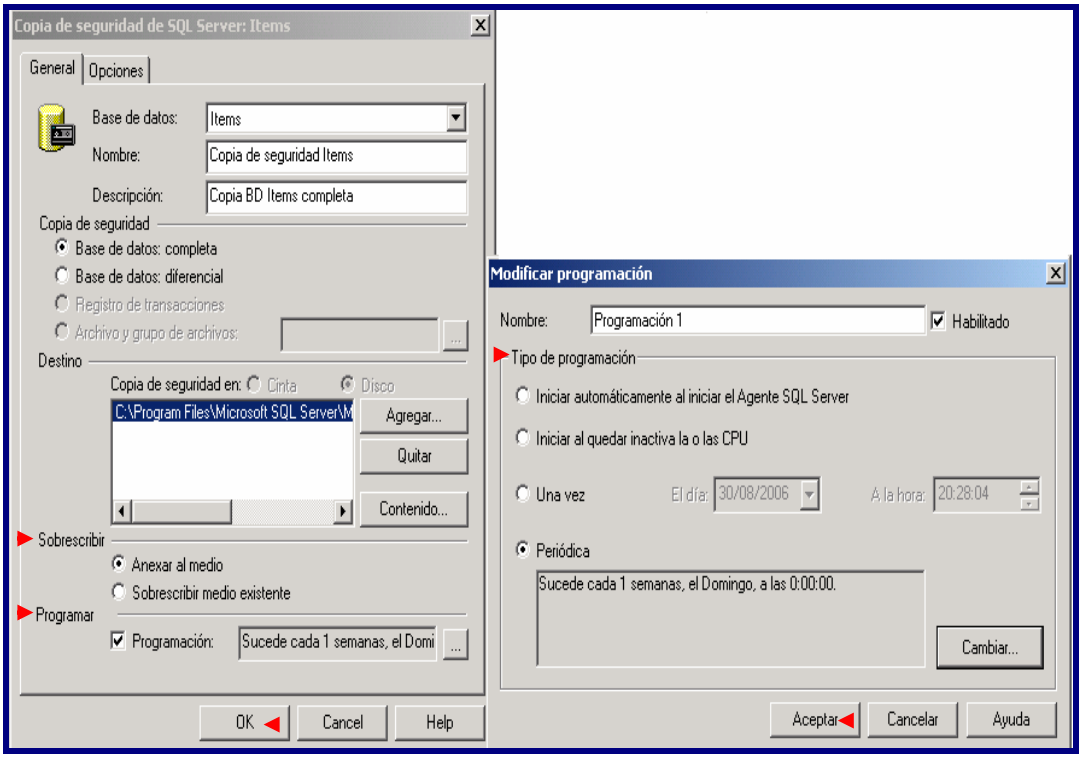
6.3.1 Realización de copias de seguridad [4]

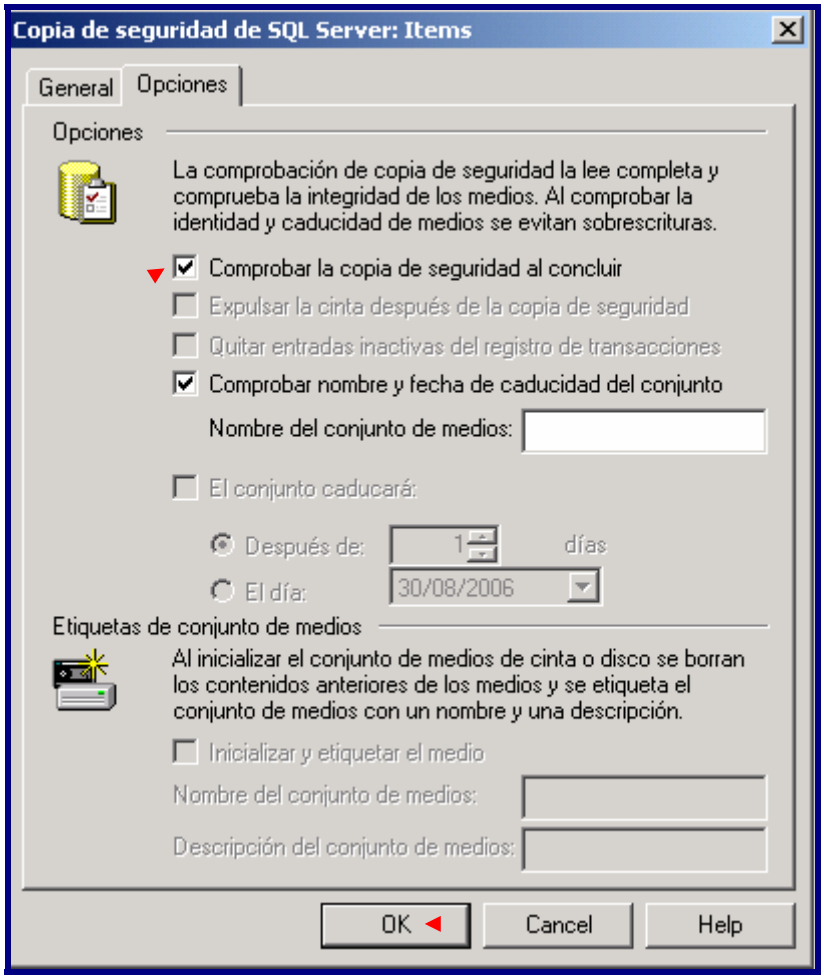
6.3.1.1 Realización de copias de seguridad por medio de interfaz gráfica



[3] Libros en pantalla de SQL Server. Versión 2000.

[4] SOLANO. Alex. Realizar copias de seguridad. http://www.ethek.com 2004 [consulta 8 de agosto de 2006].


Copia de seguridad de SQL Server: Items
General
Opciones
Base de datos:
Items
Nombre:
Copia de seguridad Items
Descripción:
Copia BD Items completa
Copia de seguridad
Base de datos: completa
Base de datos: diferencial
Registro de transacciones
Archivo y grupo de archivos:
Destino
Copia de seguridad en:
Cinta
Disco
C:\Program Files\Microsoft SQL Server\M
Agregar...
Quitar
Contenido...
Sobrescribir
Anexar al medio
Sobrescribir medio existente
Programar
Programación:
Sucede cada 1 semanas, el Domi
OK
Cancel
Help
Modificar programación
Nombre:
Programación 1
Habilitado
Tipo de programación
Iniciar automáticamente al iniciar el Agente SQL Server
Iniciar al quedar inactiva la o las CPU
Una vez
El día:
30/08/2006
A la hora:
20:28:04
Periódica
Sucede cada 1 semanas, el Domingo, a las 0:00:00.
Cambiar...
Aceptar
Cancelar
Ayuda


Copia de seguridad de SQL Server: Items
General
Opciones
Opciones
La comprobación de copia de seguridad la lee completa y comprueba la integridad de los medios. Al comprobar la identidad y caducidad de medios se evitan sobrescrituras.
Comprobar la copia de seguridad al concluir
Expulsar la cinta después de la copia de seguridad
Quitar entradas inactivas del registro de transacciones
Comprobar nombre y fecha de caducidad del conjunto
Nombre del conjunto de medios:
El conjunto caducará:
Después de:
1
días
El día:
30/08/2006
Etiquetas de conjunto de medios
Al inicializar el conjunto de medios de cinta o disco se borran los contenidos anteriores de los medios y se etiqueta el conjunto de medios con un nombre y una descripción.
Inicializar y etiquetar el medio
Nombre del conjunto de medios:
Descripción del conjunto de medios:
OK
Cancel
Help

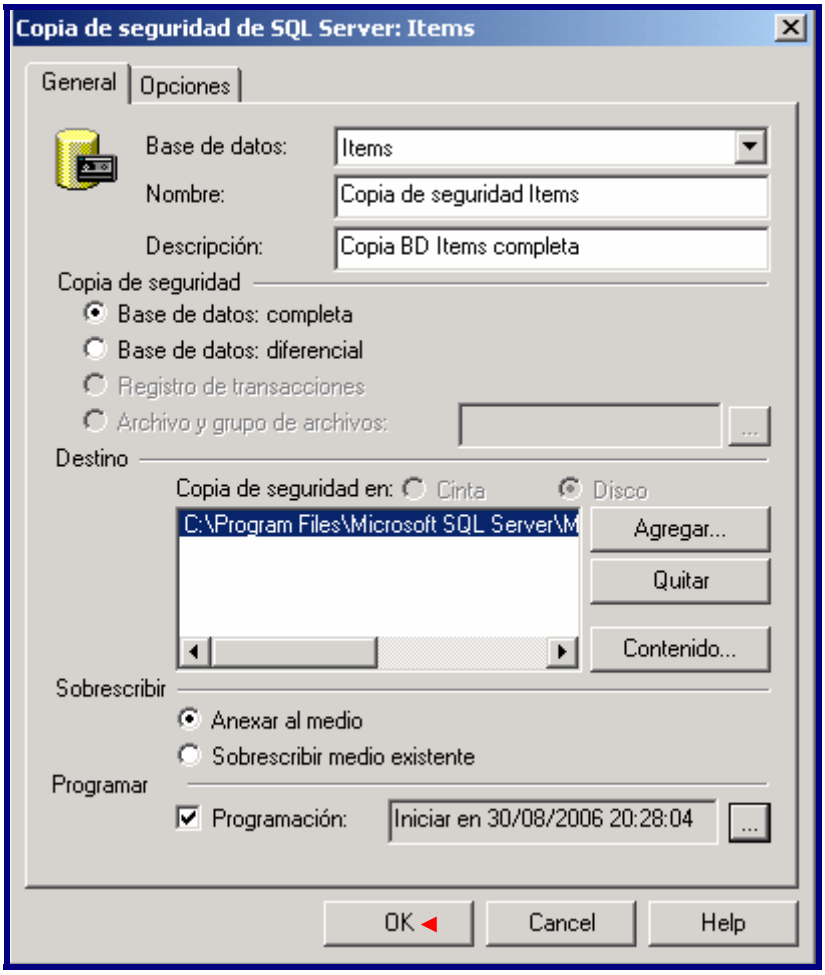


Figura 6.1: Realización de copias de seguridad utilizando interfaz gráfica (Administrador Corporativo)

Definiciones:

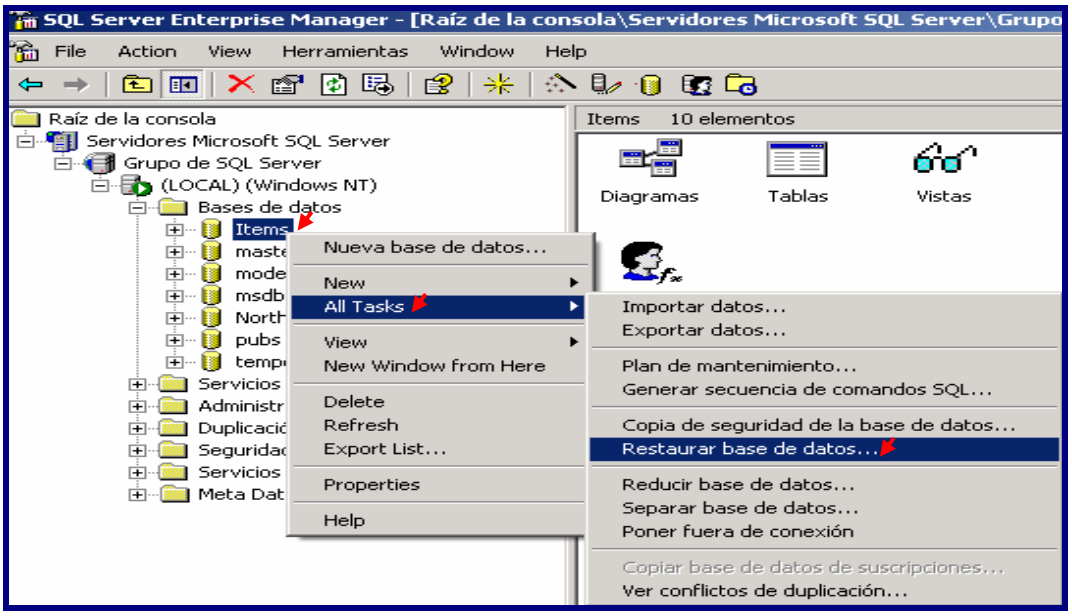
- Tipo de copia completa o diferencial. Si la copia es diferencial registra solo los cambios de la información de la base de datos realizados después de la última copia de seguridad.

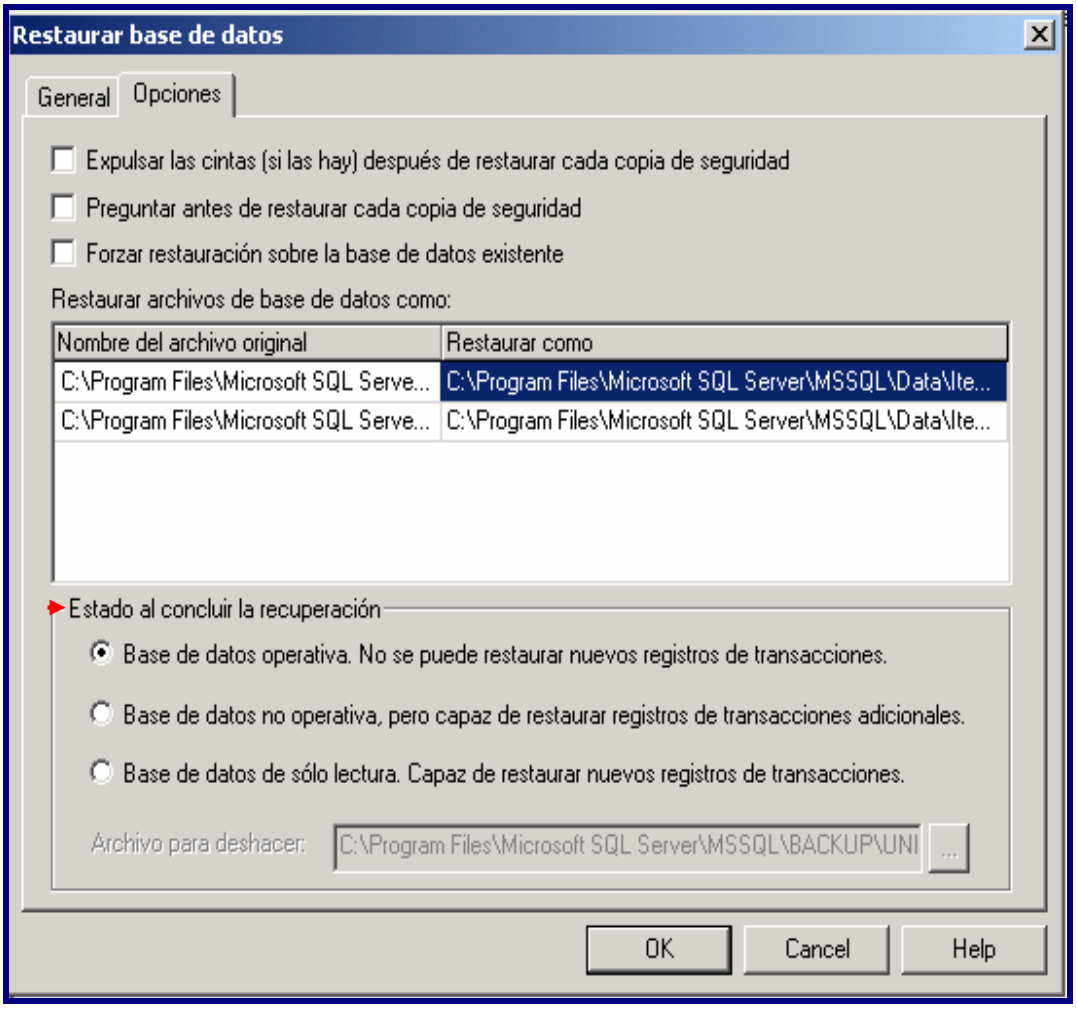
- Destino. Se coloca un nombre de fichero y un path, como se ilustró en la figura anterior

- Sobrescribir o anexar al medio. Si se sobrescribe la copia, el fichero solo contendrá la última copia de seguridad realizada. Si se anexa al medio, el fichero será incremental y contendrá todas las copias que se realicen.

- La opción programar es para realizar la ejecución de una copia de seguridad en una hora y fecha determinada.

6.3.1.2 Restaurar copias de seguridad por medio de interfaz gráfica





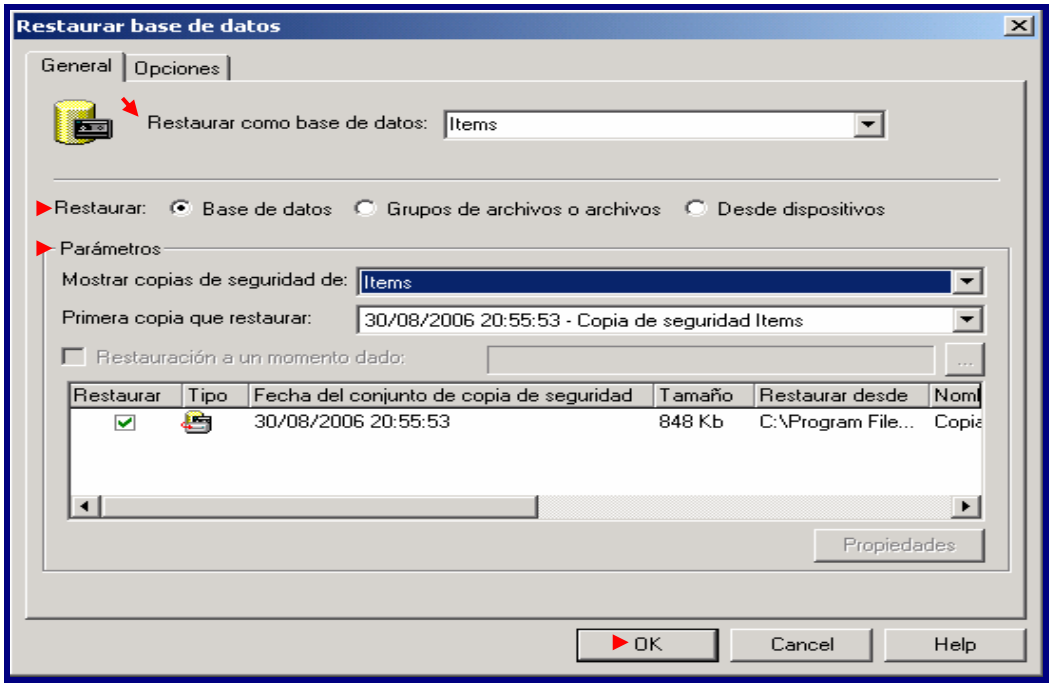


Figura 6.2: Restauración de copias de seguridad.

6.3.1.3 BACKUP

Realiza una copia de seguridad de una base de datos completa, del registro de transacciones o de uno o más archivos, o grupos de archivos, por medio del analizador de consultas. La realización de estas copias resulta útil, rápida y fácil para recuperar información. Los miembros de la función fija de servidor sysadmin o de las funciones fijas de base de datos db_owner y db_backupoperator pueden ejecutar copias de seguridad.[5]

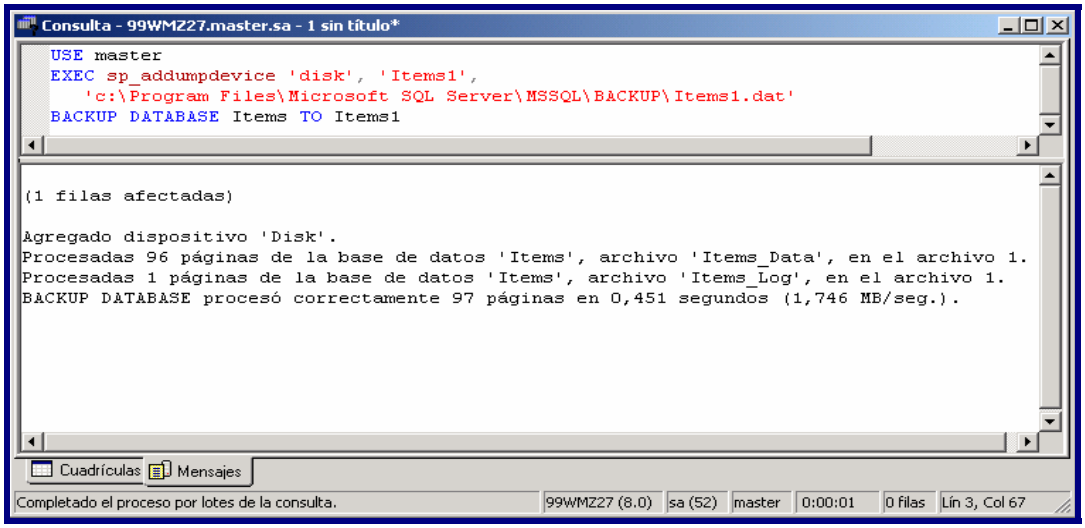


Figura 6.3: Backup completo de una base de datos.

[5] Libros en pantalla de SQL Server. Versión 2000.

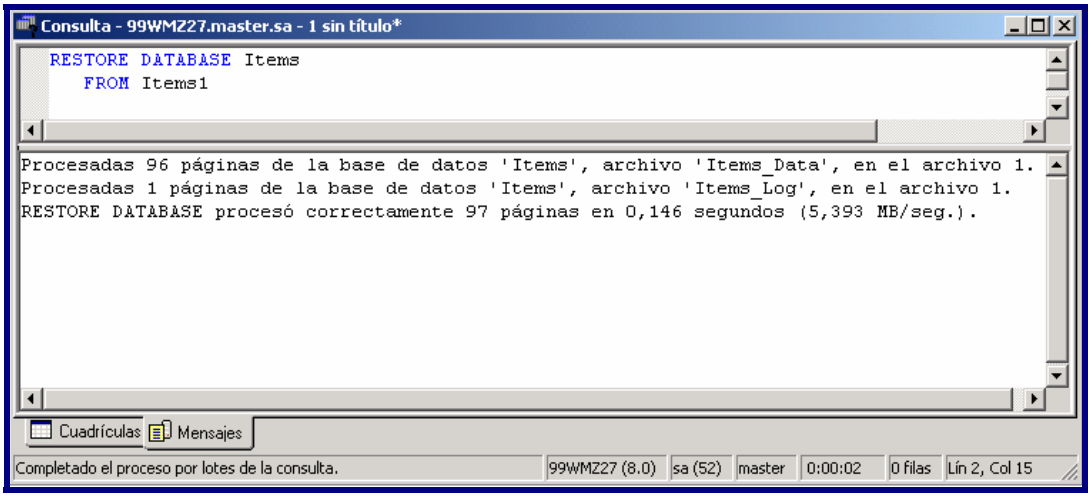


Figura 6.4: Restauración de una base de datos completa.

6.4 Conclusiones

Mientras se mantenga una adecuada asignación de contraseñas y se tenga especial cuidado con las cuentas de administrador se puede adquirir una seguridad medianamente buena. La seguridad física de un servidor es importante, por cuanto en él se deposita toda la información de una empresa, por ende se debe tener especial cuidado del espacio donde está ubicado o dispuesto este en cuanto a ventilación, o humedad.

Es importante automatizar el proceso de copias de seguridad; cuanto menor sea la interacción humana, mejor. Se debe apuntar a desarrollar un esquema que permita en cualquier momento detener el servidor y bajar el backup. Con esto la pérdida de datos será mínima (tendiendo a nula). Toda copia deberá tener registrada su fecha de caducidad para protegernos de depósitos de datos fuera de vigencia. Por otro lado, es altamente recomendable mantener un juego actualizado de copias en un almacenamiento externo al edificio.

CAPITULO 7: ILUSTRACION DEL MODULO WEB CREADO EN ASP.NET PARA EL ACCESO A LA BASE DE DATOS ITEMS

Introducción

El propósito del desarrollo de esta aplicación Web es poner en práctica lo analizado en los capítulos anteriores. La intención de este modelo es dar a conocer al usuario los diferentes mecanismos de seguridad y exigencias del gestor de base de datos para proveer de manera segura la información.

7.1 Interfaz de usuario

Figura 7.1: Presentación de la aplicación Web para el acceso a la base de datos.

Figura 7.2: Identificación de usuarios Windows o SQL Server, para el acceso a la base de datos ítems.

Figura 7.3: Opciones de inserción, modificación, eliminación y listado con la cuales se pondrá en práctica los diferentes mecanismos de seguridad.

Figura 7.4: Ingreso en la tabla artículos para los usuarios a los que se conceda el permiso de inserción, de lo contrario aparecerá una excepción con el correspondiente error.

Figura 7.5: Modificación en la tabla artículos para los usuarios a los que se conceda este permiso, de lo contrario aparecerá una excepción con el correspondiente error.

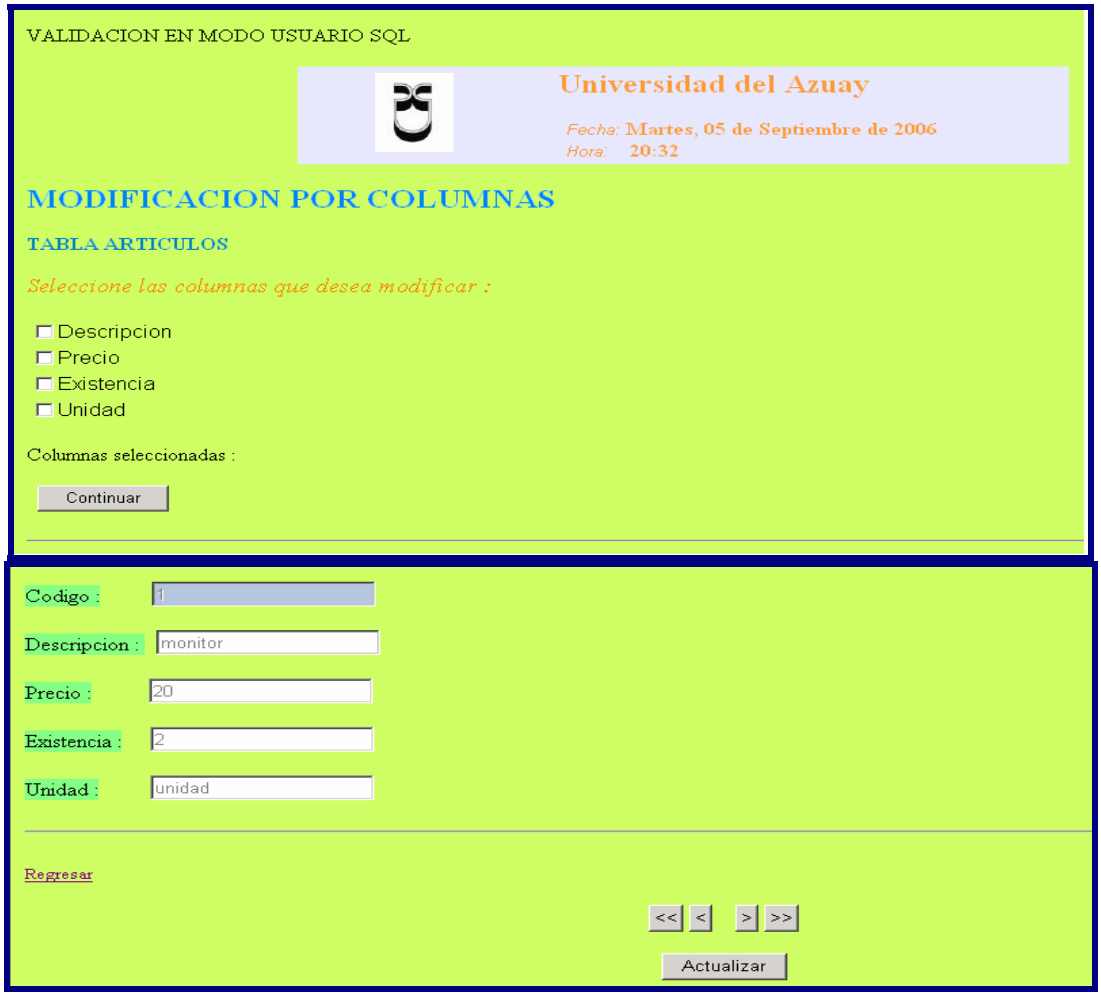


Figura 7.6: Modificación por columnas en la tabla artículos para los usuarios a los que se conceda este permiso, de lo contrario aparecerá una excepción con el correspondiente error.

Figura 7.7: Eliminación en la tabla artículos para los usuarios a los que se conceda este permiso, de lo contrario aparecerá una excepción con el correspondiente error.

Figura 7.8: Listado la tabla artículos para los usuarios a los que se conceda este permiso, de lo contrario aparecerá una excepción con el correspondiente error.

Figura 7.9: Listado por columnas en la tabla artículos para los usuarios a los que se conceda este permiso, de lo contrario aparecerá una excepción con el correspondiente error.

7.2 Componentes de la aplicación

- Sistema operativo Microsoft Windows XP Profesional.

- Microsoft SQL Server 2000 Versión 8.0. Base de datos relacional, escalable, basada en SQL.

- Microsoft Web Matrix. Mediante esta herramienta se pueden construir aplicaciones Web. Además de esto, el acceso a las Bases de Datos se lo realiza de una manera efectiva.

7.3 Tablas utilizadas

Nombre de columna	Tipo de datos	Longitud	Permitir valores nulos
Codigo	varchar	5	
Descripcion	varchar	40	
Precio	money	8	
Existencia	int	4	
Unidad	varchar	20	✓

Figura 7.10: Diseño de la tabla artículos.

7.4 Conclusiones

Con el desarrollo de esta aplicación se accedió a la base de datos ítems, implementando los mecanismos de seguridad del gestor, y así disminuyendo considerablemente los problemas de pérdida de datos fundamentales. Realizando un plan de seguridad, se identificó que usuarios en la organización pueden acceder a determinados datos, y que actividades deben realizar en la base de datos.

RECOMENDACIONES

- Realizar una planificación práctica más no rutinaria y difícil de implementar, además, conviene construir políticas de seguridad para confrontar posibles amenazas.

- Realizar planificaciones que se adapten a la forma de trabajo de las personas.

- Es fundamental la creación de inicios de sesión y usuario con el manejo de contraseñas ya que mediante esto se controla el ingreso de intrusos a cierta información, por que únicamente los usuarios creados y con ciertos privilegios pueden tener acceso a datos vitales para una empresa.

- Tener cuidado con las cuentas de administrador, que tienen acceso y privilegios totales.

- Nunca se conecte a la base de datos como un administrador o como el dueño de la base de datos, use siempre usuarios personalizados con privilegios de jerarquía.

- Renovar cada cierto tiempo las contraseñas de los usuarios ya que estas pueden ser descubiertas por personas ajenas a la institución.

- Anular inicios de sesión, usuarios y contraseñas para personas que no trabajen en la empresa, y personas que ya no tengan autorización para acceder a cierta información.

- Crear políticas para contraseñas, en donde se especifique una longitud mínima de caracteres y se restrinja contraseñas evidentes.

- No aceptar contraseñas en blanco, asegurarse de que toda cuenta posea un password.

- Los usuarios deberán finalizar la sesión cuando no estén usando su computadora o la dejan sin operar durante un cierto tiempo.

- Asignar roles y permisos para acceder y manipular información de una manera más fácil. Además mediante estos mecanismos se evita los problemas de permisos opuestos.

- Asignar privilegios de concesión, revocación y negación de permisos como listar, insertar, modificar o borrar datos. Esto es crucial para una seguridad efectiva, por cuanto la asignación de permisos ayuda alcanzar una protección adecuada para la información frente a amenazas.

- Auditar es importante. Aplicando esté mecanismo se investiga que personas y cuantas veces han intentado conectarse en forma correcta u errónea al servidor de base de datos.

- Se debe dar importancia a la seguridad física de un servidor ya que este contiene información trascendental para la empresa.

- Realizar copias de seguridad con frecuencia, dependiendo del uso de la base de datos.

- Verificar la finalización de las copias de seguridad.

- Almacenar y renovar regularmente en forma adecuada y ordenada los dispositivos de almacenamiento en caso de un desastre.

- Mantener copias de seguridad completas fuera del predio donde se encuentra el servidor de base de datos.

- Realizar comprobaciones de consistencia de las copias de seguridad con cierta frecuencia.

- Administrar las copias de seguridad con efectividad.

- Se recomienda profundizar en el tema de copias de seguridad de una base de datos, por cuanto la misma serviría como monografía para futuros estudios.

CONCLUSIONES

Toda información almacenada en una base de datos llega a tener un gran valor para las empresas. Los gestores de bases de datos tendrán que garantizar que esta información se encuentra asegurada frente a amenazas como: usuarios malintencionados que intenten leer información clasificada; ataques que deseen manipular o destruir la información; o simplemente ante las torpezas de algún usuario autorizado pero falto de conocimiento. La seguridad únicamente es eficiente y efectiva si esta es aplicada.

Existen múltiples alternativas para proteger los datos. Mediante el estudio realizado se recomiendan ciertas pautas de seguridad, como es la confidencialidad, integridad y disponibilidad de los mismos.

Los roles controlan el acceso a los servidores y objetos dentro de una base de datos, manejando permisos. Estos son de gran ayuda por que se aplican a un grupo de usuarios, inicios de sesión o funciones en lugar de aplicarlos individualmente. Por medio de la asignación de roles y permisos, únicamente las personas autorizadas pueden variar (modificar o borrar) los datos. Además de debe dar hincapié en la disponibilidad necesaria que se debe cumplir si las personas autorizadas pueden acceder a tiempo a la información, fijando una disponibilidad adecuada a los datos ya que tiene mucha gravedad la disponibilidad absoluta.

La creación y borrado de un inicio de sesión y usuario es importante para el manejo adecuado de las bases de datos, además se puede administrar la confidencialidad que se cumple cuando únicamente las personas autorizadas pueden conocer y tener acceso a una determinada información.

Mediante los privilegios de concesión, revocación y negación de acceso a los datos, se logra una convicción de que estos alcancen una protección adecuada y que la información sea accedida por personas que tienen autorización y obligaciones sobre la misma.

La auditoría ayuda a gestionar una seguridad efectiva, indagando sobre quien y cuantas veces se ha intentado conectar en forma errónea, enfrentando posibles amenazas, que pueden ser cruciales para el buen mantenimiento de la información.

Con un buen manejo y ciertas políticas adecuadas para las contraseñas y para las cuentas de administrador se puede adquirir una seguridad medianamente buena. Es importante la seguridad física de un servidor, por cuanto en él se encuentra el corazón de la empresa.

Las copias de seguridad ayudan a restaurar una base de datos en una instancia tal. Cuanto menor sea la interacción humana, mejor. Se debe apuntar a desarrollar un esquema que permita en cualquier momento detener el servidor y bajar el backup. Con esto la pérdida de datos será mínima, tendiendo a nula.

Con el desarrollo de esta aplicación se alcanzó el acceso a una base de datos, implementando los mecanismos de seguridad del gestor, así disminuyendo considerablemente los problemas de pérdida de datos fundamentales, y además realizando un plan de seguridad, el cual identificó que usuarios en la organización pueden acceder a determinados datos, y que actividades deben realizar en la base de datos.

El gestor utilizado es poderoso al hablar de seguridad de datos, pero no imposible de quebrantarlo. Si no esta bien implementada su seguridad frente a posibles amenazas, la empresa de seguro pederá información vital.

BIBLIOGRAFIA

Libros:

DOBSON. Rick. Programación de Microsoft SQL SERVER 2000 con Microsoft VISUAL BASIC.NET. Editorial Mc Graw-Hill. 1edición. España: Madrid, 2002. 617 p.

FARLEY. Marc. HSU. Jeffrey. STEARNS. Tom. Seguridad e Integridad de Datos. Editorial Mc Graw-Hill. 1edición. España: Madrid, 1998. 342 p.

FIRTMAN. Maximiliano. ASP.NET. Editorial MP Ediciones. 1edición. Argentina: Buenos Aires, 2004. 390 p.

KORTH. Henry. SILBERSCHATZ. Abraham. SUDARSHAN. S. Fundamentos de bases de datos. Editorial Mc Graw-Hill. 3edición. España: Madrid, 1998. 641 p.

Libros en pantalla de SQL Server. Versión 2000.

Artículos de Internet:

MOREA. Lucas. Seguridad de base de datos. http:// www.monografias.com. [consulta 8 de agosto de 2006]

ANCAJIMA. Robert. Grupo GESFOR OSMOS . Modelo de seguridad de SQL SERVER 7.0/2000. Perú. http:// www.informatizate.net 2004. [consulta 20 de julio de 2006]

ANCAJIMA. Robert. Grupo GESFOR OSMOS . Seguridad en SQL SERVER 2000. Perú. http:// www.informatizate.net 2004. [consulta 20 de julio de 2006]

Seguridad de base de datos. http://www.hospedajeydominios.com/mambo/Seguridad de Base de Datos.htm. [consulta 8 de agosto de 2006]

SOLANO. Alex. Realizar copias de seguridad. http://www.ethek.com 2004 [consulta 8 de agosto de 2006]

SQL Server. Seguridad y Performance (rendimiento). http://www.helpdna.net/SQL Server FAQ's, Resolución de problemas frecuentes con SQL Server y todos sus componentes.htm [consulta 12 de agosto de 2006]

ANEXOS

Anexo1: Diseño de monografía

Anexo2: CD (Aplicación Web, "Estudio práctico sobre la seguridad de los datos con el gestor de base de datos Microsoft Sql Server 2000")