Administración de la seguridad

Contenido

Notas para el instructor

Este módulo proporciona a los alumnos los detalles de implementación de la seguridad. Comienza con la descripción de cómo se configura el modo de autenticación en un servidor y cómo se concede acceso a usuarios y grupos de Microsoft® Windows® 2000 y a usuarios de Microsoft SQL Server™ 2000. La siguiente sección describe cómo se asignan cuentas de inicio de sesión a usuarios y funciones, y cómo se asignan permisos a usuarios y funciones. En este módulo también se trata cómo se lleva a cabo la administración de la seguridad con SQL Server y la administración de la seguridad de la aplicación. Concluye con la descripción del modo en que se administra la seguridad de SQL Server en un entorno empresarial.

Este módulo tiene tres prácticas. En la primera práctica, los alumnos establecerán un modo de autenticación y concederán acceso a SQL Server a cuentas de inicio de sesión. En la segunda práctica, los alumnos asignarán cuentas de inicio de sesión a cuentas de usuario y funciones, y asignarán permisos a las cuentas de usuario y las funciones. En la tercera práctica, los alumnos crearán una función de aplicación, le asignarán permisos y la comprobarán. Los alumnos también crearán una vista y un procedimiento almacenado que proporciona acceso a una tabla sin conceder a los usuarios acceso directo a la tabla.

Después de completar este módulo, los alumnos serán capaces de:

- Implementar el Modo de autenticación y el Modo mixto de Microsoft Windows.
- Asignar cuentas de inicio de sesión a cuentas de usuario y funciones de la base de datos.
- Asignar permisos a cuentas de usuario y funciones.
- Administrar la seguridad en SQL Server.
- Administrar la seguridad de la aplicación.
- Administrar la seguridad de SQL Server en un entorno empresarial.

Introducción

Objetivo del tema
Proporcionar una introducción a los temas y objetivos del módulo.

Explicación previa
En este módulo aprenderá la seguridad de SQL Server, en el nivel de servidor y en el de base de datos.

- Implementación del modo de Autenticación
- Asignación de cuentas de inicio de sesión a usuarios y funciones
- Asignación de permisos a usuarios y funciones
- Administración de la seguridad en SQL Server
- Administración de la seguridad de la aplicación
- Administración de la seguridad de SQL Server en la empresa

Este módulo proporciona los detalles de implementación de la seguridad. Aprenderá a configurar un modo de autenticación y a conceder acceso a usuarios y grupos de Microsoft® Windows® 2000 y a usuarios de Microsoft SQL Server™ 2000. Aprenderá cómo se asignan cuentas de inicio de sesión a usuarios y funciones, y cómo se asignan permisos a usuarios y funciones. Por último, aprenderá a administrar la seguridad mediante SQL Server, la seguridad de la aplicación y la seguridad en un entorno empresarial.

Después de realizar esta práctica, usted será capaz de:

- Implementar el Modo de autenticación y el Modo mixto de Microsoft Windows.
- Asignar cuentas de inicio de sesión a cuentas de usuario y funciones de la base de datos.
- Asignar permisos a cuentas de usuario y funciones.
- Administrar la seguridad en SQL Server.
- Administrar la seguridad de la aplicación.
- Administrar la seguridad de SQL Server en un entorno empresarial.

Implementación del modo de Autenticación

Objetivo del tema
Presentar el tema de la autenticación.

Explicación previa
La seguridad de SQL Server utiliza dos modos de autenticación diferentes: Modo de autenticación y Modo mixto de Windows.

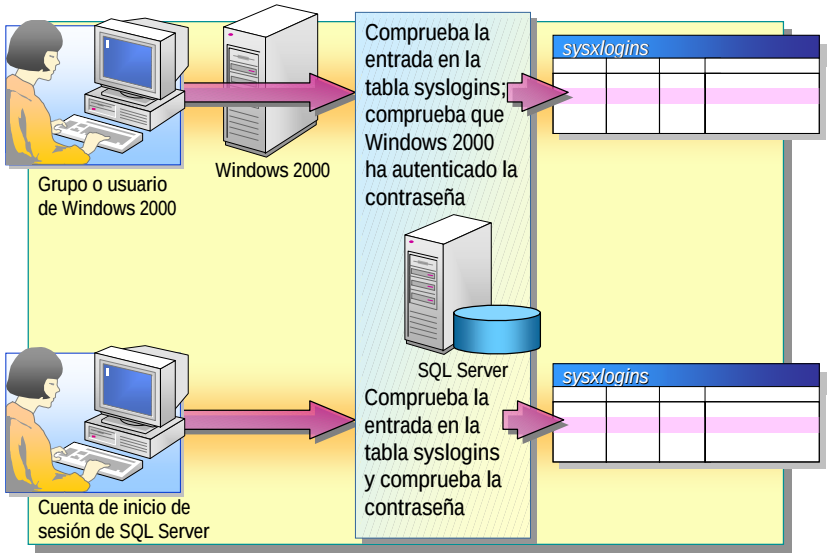
- Proceso de autenticación
- Elección del modo de autenticación
- Autenticación mutua con Kerberos
- Representación y delegación
- Cifrado
- Pasos para implementar un Modo de autenticación
- Creación de cuentas de inicio de sesión
- Configuración de cuentas de inicio de sesión

Se puede proteger SQL Server 2000 mediante la implementación del Modo de autenticación o del Modo mixto de Windows. En esta sección se describe el proceso de autenticación de cada modo, los pasos que hay que seguir para implementar la autenticación y cómo crear cuentas de inicio de sesión.

Proceso de autenticación

Objetivo del tema
Presentar cómo se procesan las cuentas de inicio de sesión.

Explicación previa
SQL Server puede delegar en Windows 2000 la autenticación de las cuentas de inicio de sesión o puede autenticar las cuentas de inicio de sesión por sí mismo.



Sugerencia
Pregunta: ¿Qué pueden hacer los usuarios después de ser autenticados si sus cuentas de inicio de sesión no tienen permisos en ninguna base de datos?

Respuesta: Pueden tener acceso al servidor y a algunos objetos del sistema, pero no pueden tener acceso a ninguna base de datos de usuario.

SQL Server puede delegar en Windows 2000 la autenticación de las cuentas de inicio de sesión o puede autenticar las cuentas de inicio de sesión por sí mismo.

Cómo procesa SQL Server las cuentas de inicio de sesión que autentica Windows 2000

A continuación se describe cómo procesa SQL Server las cuentas de inicio de sesión que autentica Windows 2000:

- Cuando un usuario se conecta con SQL Server, el cliente abre una conexión de confianza con SQL Server, que pasa las credenciales de seguridad de Windows 2000 del usuario a SQL Server.

  Como el cliente tiene abierta una conexión de confianza, SQL Server sabe que Windows 2000 ya ha validado la cuenta de inicio de sesión.

- Si SQL Server encuentra la cuenta de usuario o de grupo de Windows 2000 del usuario en la lista de cuentas de inicio de sesión de SQL Server que se encuentra en la tabla del sistema **sysxlogins**, acepta la conexión.

  SQL Server no necesita volver a validar la contraseña porque Windows 2000 ya la ha validado.

Nota SQL Server no reconocerá a los usuarios o grupos que usted haya eliminado y vuelto a crear en Windows 2000. Las cuentas de Windows 2000 se identifican internamente mediante un identificador de seguridad único (SID) que no se vuelve a utilizar jamás. Debe eliminar la cuenta de SQL Server y agregarla de nuevo, ya que SQL Server utiliza el SID para identificarla.

- Si varios equipos que ejecutan SQL Server participan en un dominio o en grupo de dominios de confianza, el inicio de sesión en un solo dominio de red es suficiente para permitir el acceso a todos los equipos que ejecutan SQL Server.

Nota Las utilidades de línea de comandos de SQL Server aceptan opciones que le permiten conectarse a través de una conexión de confianza.

Cómo procesa SQL Server las cuentas de inicio de sesión que él mismo autentica

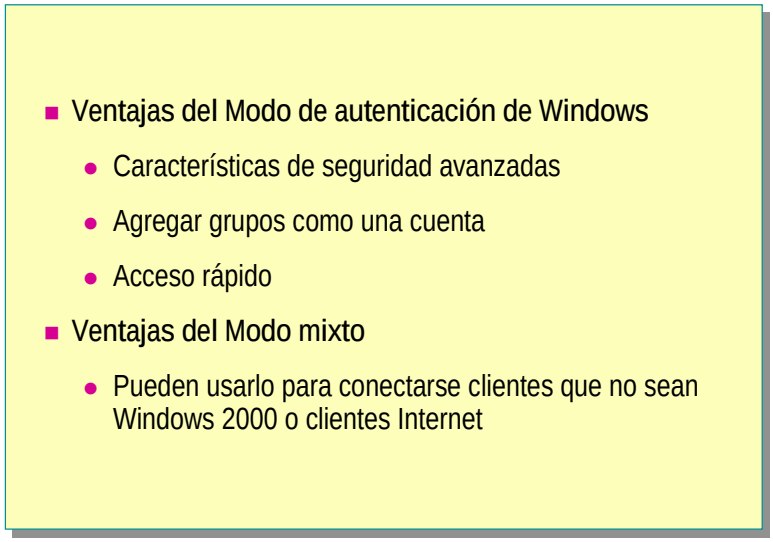
SQL Server emplea los siguientes pasos para procesar las cuentas de inicio de sesión que él mismo autentica:

- Cuando un usuario conecta con una cuenta de inicio de sesión y contraseña de SQL Server, éste comprueba si existe la cuenta de inicio de sesión en la tabla **sysxlogins** y si la contraseña especificada coincide con la contraseña registrada.
- Si SQL Server no tiene configurada una cuenta de inicio de sesión para el usuario, la autenticación falla y la conexión se rechaza.

Elección del modo de autenticación

Objetivo del tema
Comparar las ventajas de los dos modos de autenticación de seguridad.

Explicación previa
Las necesidades de seguridad del servidor y de los entornos de red determinarán el modo de autenticación que se deba utilizar con SQL Server.



Las necesidades de seguridad del servidor y de los entornos de red determinarán el modo de autenticación que se deba utilizar con SQL Server. Puede utilizar el Administrador corporativo de SQL Server para establecer el modo de autenticación del servidor.

Ventajas del Modo de autenticación de Windows

Utilice el modo de Autenticación de Windows en entornos de red en los que todos los clientes admiten conexiones de confianza.

La autenticación de Windows ofrece varias ventajas frente a la autenticación de SQL Server, ya que:

- Proporciona más características, como la validación y el cifrado seguros de contraseñas, la auditoría, la caducidad de las contraseñas, la longitud mínima de la contraseña y el bloqueo de las cuentas después de contraseñas no válidas.
- Permite agregar grupos de usuarios a SQL Server con sólo agregar una única cuenta de inicio de sesión.
- Permite que los usuarios tengan acceso a SQL Server rápidamente, sin necesidad de recordar otra cuenta de inicio de sesión y contraseña.

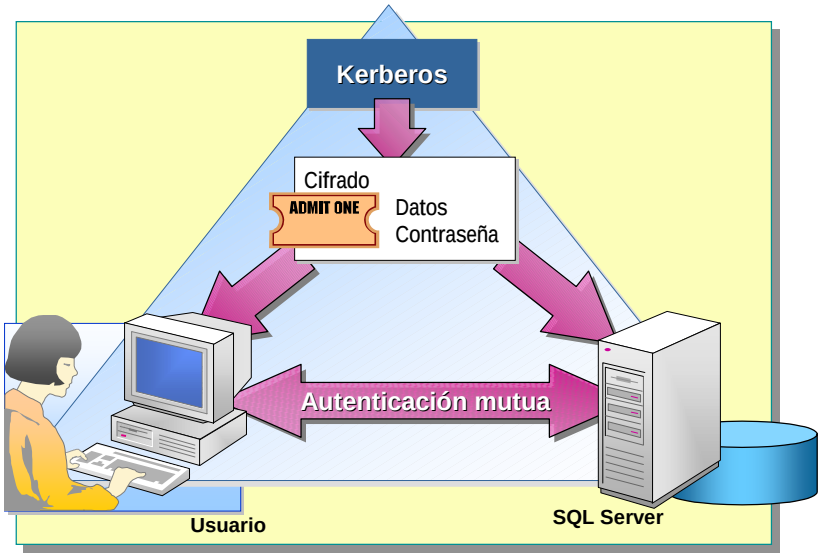
Ventajas del Modo mixto

El Modo mixto, y los mecanismos de autenticación de SQL Server en particular, permiten que clientes que no utilicen Windows 2000, clientes de Internet y grupos de clientes mixtos se conecten a SQL Server.

Autenticación mutua con Kerberos

Objetivo del tema
Presentar los principios básicos del protocolo Kerberos.

Explicación previa
Kerberos es el principal protocolo de seguridad para la autenticación en Windows 2000.



Kerberos es el principal protocolo de seguridad para la autenticación en un dominio de Windows 2000. Define el modo en que los clientes interactúan con los servicios de autenticación de red.

El protocolo Kerberos comprueba tanto la identidad del usuario como los servicios de red. Esta doble autenticación es conocida como autenticación mutua. SQL Server 2000 utiliza Kerberos para posibilitar la autenticación mutua entre el cliente y el servidor.

Cuando se inicia sesión, Windows 2000 ubica un servidor de servicio de directorio Active Directory™ y un servicio de autenticación Kerberos. El servicio Kerberos emite un vale de concesión de vales (TGT, *ticket-granting ticket*) que contiene datos cifrados que confirman la identidad del usuario al Centro de distribución de claves (KDC, *Key Distribution Center*). Cuando el usuario solicita acceso a un servicio, éste envía el TGT al KDC, que enviará un vale de sesión al usuario para el servicio solicitado. El usuario presenta entonces el vale de sesión, que confirma su identidad, al servicio.

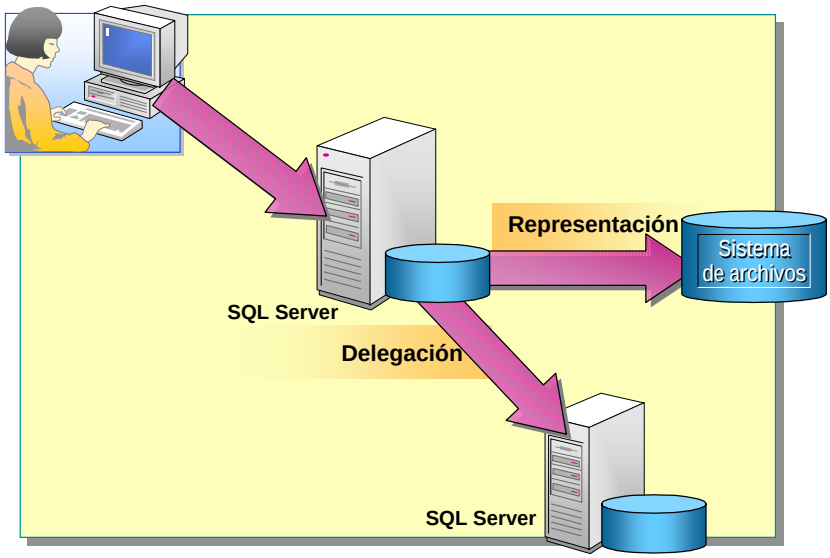
Si es necesaria la autenticación mutua, el servidor responde con un mensaje cifrado para identificarse a sí mismo. El vale de sesión se puede volver a utilizar para tener acceso al mismo servicio hasta que caduque, lo cual elimina la necesidad de que el KDC esté emitiendo constantemente vales repetidos. El tiempo de caducidad lo determina el KDC, pero normalmente no dura más de ocho horas, la duración de un inicio de sesión normal.

Nota Una autenticación correcta con Kerberos requiere que tanto el sistema cliente como el servidor que ejecutan el servicio deseado tengan instalado el sistema operativo Windows 2000 y que utilicen la biblioteca de red de sockets TCP/IP, y que el nombre principal de servicio (SPN) esté adecuadamente configurado para el servicio de SQL Server.

Representación y delegación

Objetivo del tema
Explicar los conceptos de representación y delegación.

Explicación previa
La representación y la delegación permiten que SQL Server utilice las credenciales de seguridad del cliente original al tener acceso a recursos de los servidores locales o remotos.



La representación y la delegación permiten que SQL Server utilice las credenciales de seguridad del cliente original al tener acceso a recursos de los servidores locales o remotos.

Representación

En ciertas circunstancias, SQL Server debe presentar una identidad del cliente a los recursos a los que tiene acceso en representación del cliente, por ejemplo, al sistema de archivos. El servidor puede representar el contexto de seguridad del cliente para permitir que se realicen comprobaciones de acceso o autenticaciones en la identidad del cliente, lo que garantiza que el acceso a los recursos ni se restringe ni se amplía más allá de los propios permisos del cliente.

Delegación

SQL Server 2000 también admite la delegación, que es la capacidad de conectarse a varios servidores y, en cada cambio de servidor, conservar las credenciales del cliente original. Por ejemplo, si un usuario (NWTraders\SQLAdmin20) se conecta a ServerA, que se conecta a ServerB a su vez, ServerB sabe que la identidad de seguridad de la conexión es NWTraders\SQLAdmin20.

Para utilizar la delegación, todos los servidores a los que se conecte el usuario deben ejecutar Windows 2000 con la compatibilidad con Kerberos habilitada, y el usuario debe utilizar Active Directory.

Configuración de Active Directory para la delegación

Para que la delegación funcione, se deben establecer las siguientes opciones de cuenta en Active Directory:

- La cuenta es importante y no se puede delegar

  No debe seleccionar esta opción para la delegación de la solicitud del usuario.

- Se confía en la cuenta para su delegación

  Debe seleccionar esta opción para la cuenta del servicio de SQL Server.

- Este equipo está aprobado para su delegación

  Debe seleccionar esta opción para el equipo que ejecuta SQL Server.

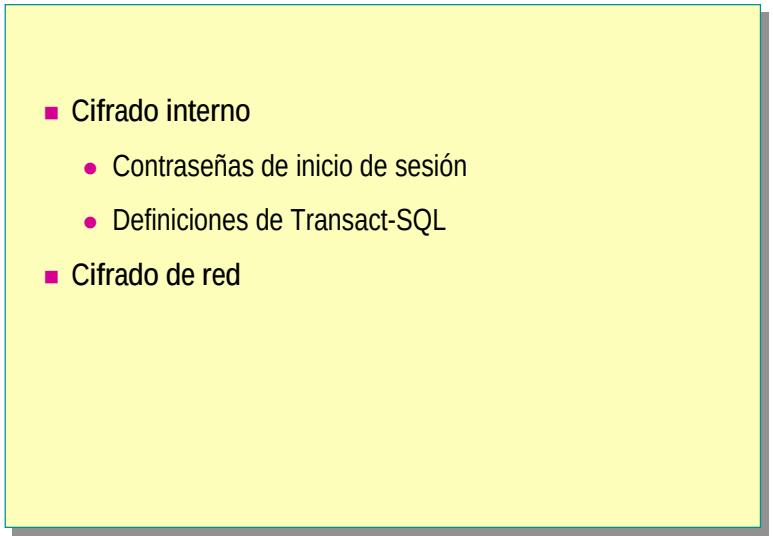
Configuración de SQL Server para la delegación

Para utilizar la delegación de la cuenta de seguridad, SQL Server debe tener asignado un SPN por el administrador de dominio de cuentas de Windows 2000 a la cuenta de servicio de SQL Server y estar utilizando el Protocolo de control de transporte/Protocolo Internet (TCP/IP). Si el servicio de SQL Server se está ejecutando bajo la cuenta LocalSystem, al inicio del servicio el servicio de SQL Server registra automáticamente un SPN y lo deja de registrar cuando se cierra SQL Server.

Cifrado

Objetivo del tema
Presentar las funciones de cifrado disponibles en SQL Server.

Explicación previa
El cifrado es un método de protección de la información confidencial que se crea pasando los datos a un formato ilegible.



El cifrado es un método de protección de la información confidencial que se crea pasando los datos a un formato ilegible. Garantiza la seguridad de los datos, incluso si se ven directamente.

Cifrado interno

SQL Server puede cifrar:

- Las contraseñas de inicio de sesión almacenadas en SQL Server.

  Las contraseñas de inicio de sesión almacenadas en tablas del sistema que siempre están cifradas. Esto impide que todos los usuarios, incluidos los administradores del sistema, vean ninguna contraseña, incluso la suya propia.

- Las definiciones de Transact-SQL.

  Las definiciones de procedimientos almacenados, los desencadenadores y las vistas almacenadas en la tabla de sistema **syscomments** se pueden cifrar.

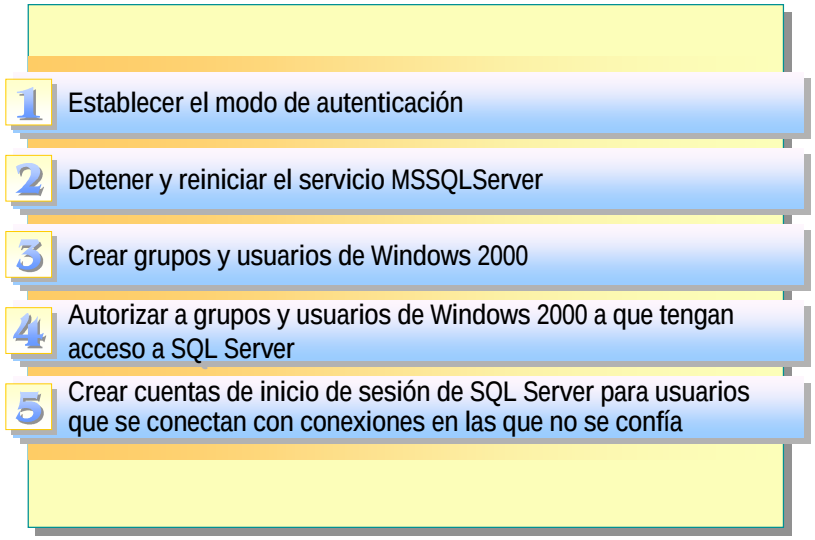
Cifrado de red

SQL Server permite que se cifren los datos que se envían entre el cliente y el servidor. Esto garantiza que cualquier aplicación o usuario que intercepte los paquetes de datos de la red no pueda ver los datos confidenciales, por ejemplo, contraseñas enviadas a través de la red cuando los usuarios inician sesión en SQL Server, o datos personales que contienen información acerca de los salarios.

Pasos para implementar un Modo de autenticación

Objetivo del tema
Describir los pasos necesarios para implementar la autenticación.

Explicación previa
Para implementar la autenticación, debe realizar las tareas siguientes.



Para implementar la autenticación, debe realizar las tareas siguientes desde una cuenta de administración del sistema. Para el Modo de autenticación de Windows, siga los pasos del 1 al 4; para el Modo mixto, siga los pasos del 1 al 5:

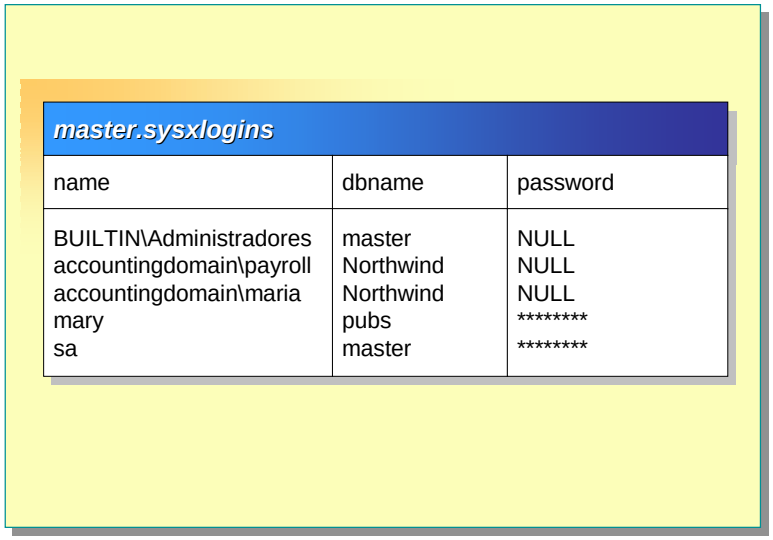
1. Utilice el Administrador corporativo de SQL Server para establecer el modo de autenticación de SQL Server.
2. Detenga y vuelva a iniciar el servicio MSSQLServer para que surta efecto la opción de seguridad.
3. Cree los grupos y usuarios de Windows 2000 que tengan autorización de acceso a SQL Server a través de conexiones de confianza.

   Si no tiene permisos para administrar grupos y usuarios en Windows 2000, haga que un administrador de Windows 2000 realice esta tarea.
4. Utilice el Administrador corporativo de SQL Server para conceder acceso a SQL Server a los grupos y usuarios de Windows 2000.
5. Utilice el Administrador corporativo de SQL Server para crear cuentas de inicio de sesión de SQL Server a fin de que los usuarios se conecten a conexiones en las que no se confía.

Creación de cuentas de inicio de sesión

Objetivo del tema
Presentar el proceso de la creación de cuentas de inicio de sesión.

Explicación previa
Puede crear cuentas de inicio de sesión a partir de usuarios y grupos de Windows 2000 existentes, o puede crear nuevas cuentas de inicio de sesión en SQL Server.



Puede crear cuentas de inicio de sesión a partir de usuarios y grupos de Windows 2000 existentes, o puede crear nuevas cuentas de inicio de sesión en SQL Server. También puede utilizar una de las cuentas de inicio de sesión predeterminadas. Las cuentas de inicio de sesión están almacenadas en la tabla del sistema **master.sysxlogins**. Cuando se agrega una cuenta de inicio de sesión a SQL Server, se le suele asignar una base de datos predeterminada. La asignación de una base de datos predeterminada a una cuenta de inicio de sesión no crea una cuenta de usuario en esa base de datos, sino que establece el contexto predeterminado para las acciones que el usuario realice.

Agregar una cuenta de inicio de sesión de Windows 2000 a SQL Server

Puede utilizar el Administrador corporativo de SQL Server o el procedimiento almacenado del sistema **sp_grantlogin** para permitir que una cuenta de usuario o de grupo de Windows 2000 se conecte con SQL Server. Sólo los administradores del sistema o de seguridad pueden ejecutar **sp_grantlogin**.

Para su información
Windows 2000 permite nombres de usuario más largos que SQL Server.

El nombre especificado al crear una cuenta es el nombre del usuario o grupo de Windows 2000 que se va a agregar. Este nombre se debe cualificar con un nombre de dominio de Windows 2000. El límite para los nombres de usuario o de grupo combinados con el dominio en SQL Server es de 128 caracteres Unicode.

Al agregar cuentas de inicio de sesión de Windows 2000 a SQL Server, tenga en cuenta los siguientes hechos y directrices:

- Como SQL Server tiene una sola cuenta de inicio de sesión para un grupo de Windows 2000, no es necesario realizar cambios en SQL Server cuando cambia la pertenencia a los grupos de Windows 2000. Esto impide que haya objetos huérfanos (objetos cuyo propietario es un usuario que ya no existe en SQL Server), siempre y cuando no elimine el grupo.
- La eliminación de un grupo o usuario de Windows 2000 no elimina ese grupo o usuario en SQL Server. Esto impide que haya objetos huérfanos.

  Cuando quita usuarios o grupos de Windows 2000, primero debe quitarlos de Windows 2000 para denegar el acceso desde la red. Después, debe quitarlos de SQL Server.
- Agregue una cuenta de inicio de sesión para una cuenta de grupo de Windows 2000 si todos los miembros del grupo se van a conectar con SQL Server.
- Agregue una cuenta de inicio de sesión para una cuenta de usuario de Windows 2000 si el usuario no es miembro de un grupo al que se le puede conceder permiso de forma colectiva.
- Incluso si los usuarios inician sesiones en SQL Server como miembros de grupos de Windows 2000, SQL Server sigue conociendo las identidades de los usuarios. La función **SUSER_SNAME()** devuelve los nombres de la cuenta de inicio de sesión y el dominio de los usuarios cuando los usuarios son miembros de un grupo de Windows 2000.
- SQL Server sólo admite la especificación de cuentas de Windows con el formato DOMINIO\nombreUsuario. SQL Server no permite que se agreguen cuentas de Windows mediante la utilización de nombres principales de usuario (UPN), nombres con el formato *alguien*@microsoft.com.

En la tabla siguiente se enumeran otros procedimientos almacenados del sistema que se pueden utilizar para administrar cuentas de inicio de sesión de Windows 2000. Sólo los administradores del sistema o de seguridad pueden ejecutar estos procedimientos almacenados del sistema.

Procedimiento almacenado de sistema	Descripción
sp_revokelogin	Quita de SQL Server las entradas de cuenta de inicio de sesión de un usuario o grupo de Windows 2000.
sp_denylogin	Impide que un usuario o grupo de Windows 2000 se conecte con SQL Server.

También se puede denegar el acceso a las cuentas de inicio de sesión mediante **Propiedades de inicio de sesión**.

Agregar una cuenta de inicio de sesión de SQL Server

Puede utilizar el Administrador corporativo de SQL Server o el procedimiento almacenado del sistema **sp_addlogin** para crear una cuenta de inicio de sesión de SQL Server. Sólo los administradores del sistema o de seguridad pueden ejecutar **sp_addlogin**.

Al crear una nueva cuenta de inicio de sesión de SQL Server se agrega un registro a la tabla **sysxlogins** de la base de datos **master**.

Las cuentas de inicio de sesión y las contraseñas de SQL Server pueden contener hasta 128 caracteres y pueden incluir letras, símbolos y números. Sin embargo, las cuentas de inicio de sesión no pueden:

- Contener el carácter de barra invertida.
- Ser una cuenta de inicio de sesión reservada; por ejemplo, **sa**.
- Ser NULL o una cadena de texto vacía ("").

Los usuarios pueden cambiar sus propias contraseñas en cualquier momento con el procedimiento almacenado del sistema **sp_password**. Los administradores del sistema pueden cambiar la contraseña de cualquier usuario mediante **sp_password** con NULL como contraseña antigua. Los usuarios y administradores del sistema también pueden cambiar las contraseñas con **Propiedades de inicio de sesión**.

Cuentas de inicio de sesión predeterminadas

Para su información
Se puede utilizar el prefijo BUILTIN sólo con los nombres de grupo de Windows 2000 predeterminados.

SQL Server tiene dos cuentas de inicio de sesión predeterminadas: **BUILTIN\Administradores** y **sa**.

BUILTIN\Administradores se proporciona como cuenta de inicio de sesión predeterminada para todos los administradores de Windows 2000. Tiene todos los derechos sobre SQL Server y todas las bases de datos.

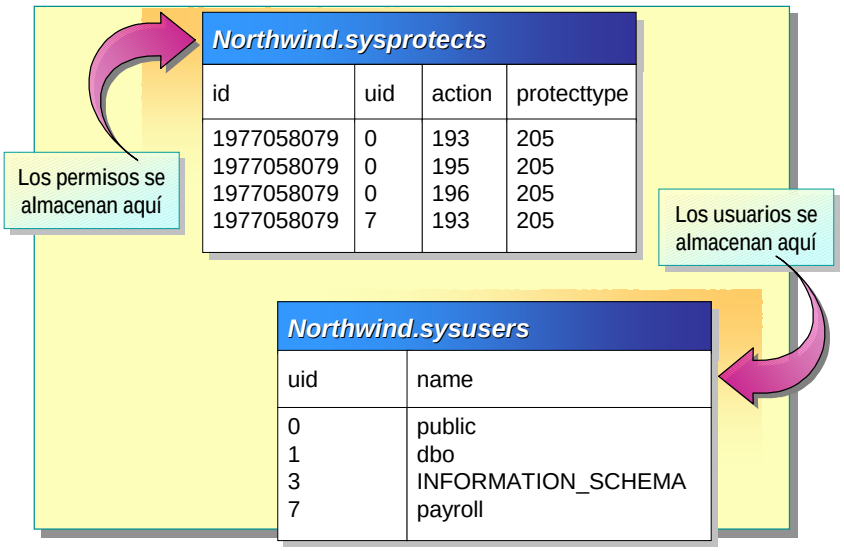
El administrador del sistema (**sa**) es una cuenta de inicio de sesión especial que tiene todos los derechos sobre SQL Server y todas las bases de datos. Se proporciona para la compatibilidad con versiones anteriores y no se debe utilizar rutinariamente. Esta cuenta se habilita sólo cuando SQL Server utiliza el Modo mixto de autenticación.

Nota Cuando se utiliza el Modo mixto, es necesario comprobar que la contraseña **sa** no está en blanco; ya que si fuera así, SQL Server no sería seguro.

Asignación de cuentas de inicio de sesión a usuarios y funciones

Objetivo del tema
Mostrar la relación entre las tablas del sistema.

Explicación previa
Después de agregar cuentas de inicio de sesión a SQL Server, puede asignarlas a cuentas de usuario o funciones en cada una de las bases de datos a las que los usuarios tengan acceso.



Para su información
A continuación se muestran los valores de las columnas de la tabla Northwind.sysprotects.
Id.:
1977058079 = tabla Employees
Acción:
193 = SELECT
195 = INSERT
196 = DELETE
197 = UPDATE
protecttype:
204 = GRANT_W_GRANT
205 = GRANT
206 = REVOKE

Sugerencia
Recorra este ejemplo con los alumnos y, después, pídales que identifiquen el modo en que un miembro de un grupo global de Windows 2000, payroll, puede tener acceso a la tabla employees.

Después de agregar cuentas de inicio de sesión a SQL Server, puede asignarlas a cuentas de usuario o funciones en cada una de las bases de datos a las que los usuarios tengan acceso.

La tabla del sistema **sysusers** de una base de datos contiene una fila por cada usuario de Windows 2000, grupo de Windows 2000, usuario de SQL Server o función de la base de datos. Los permisos se aplican a las entradas de la tabla **sysusers** y se almacenan en la tabla **sysprotects** de la base de datos actual.

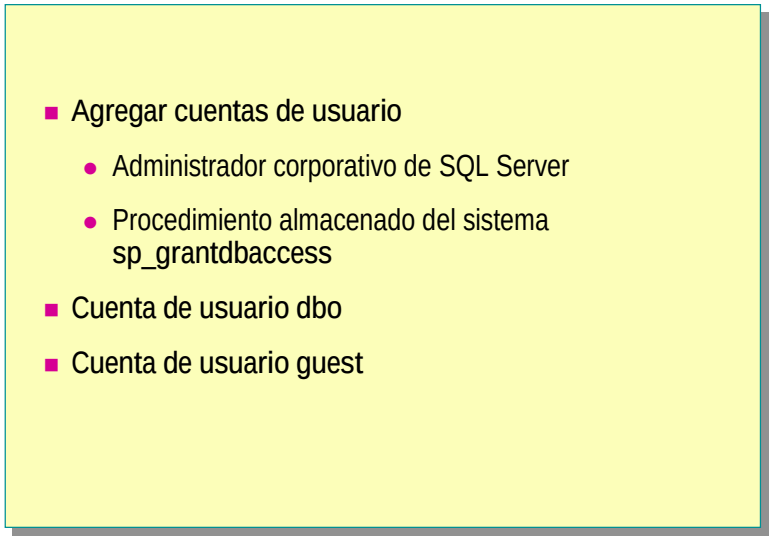
Por ejemplo, en la ilustración anterior, se les ha concedido a los miembros del grupo de Windows 2000 **payroll** acceso a la base de datos **Northwind**. Después de conectarse con SQL Server, el usuario:

- Tiene una conexión validada con SQL Server, porque su cuenta de usuario de Windows 2000 está en el grupo global de Windows 2000 **payroll**.
- No tiene permisos individuales en la base de datos **Northwind**, sólo tiene permisos el grupo global **payroll**.

Asignación de cuentas de inicio de sesión a cuentas de usuario

Objetivo del tema
Presentar las cuentas de usuario de la base de datos.

Explicación previa
Para tener acceso a una base de datos, una cuenta de inicio de sesión puede utilizar una cuenta de usuario predeterminada o una cuenta de usuario asignada.



Para tener acceso a una base de datos, una cuenta de inicio de sesión puede utilizar una cuenta de usuario asignada o una predeterminada.

Agregar cuentas de usuario

Para agregar una cuenta de usuario a una base de datos, puede utilizar el Administrador corporativo de SQL Server o ejecutar el procedimiento almacenado del sistema **sp_grantdbaccess**. También puede agregar cuentas de usuario a bases de datos al crear cuentas de inicio de sesión. Sólo los propietarios de la base de datos o los administradores del acceso a la base de datos pueden ejecutar el procedimiento **sp_grantdbaccess**.

En el árbol de consola del Administrador corporativo de SQL Server, cada base de datos tiene una carpeta Users. Aquí se muestra la lista de usuarios actual de la base de datos y se le permite agregar y quitar usuarios y personalizar sus propiedades.

En la tabla siguiente se enumeran otros procedimientos almacenados del sistema que se pueden utilizar para administrar el acceso a la base de datos.

Procedimiento almacenado de sistema	Descripción
sp_revokedbaccess	Quita una cuenta de seguridad de la base de datos actual.
sp_change_users_login	Cambia la relación entre una cuenta de inicio de sesión de SQL Server y un usuario de SQL Server en la base de datos actual.

Cuenta de usuario dbo

El inicio de sesión **sa** y los miembros de la función **Administradores del sistema** (**sysadmin**) están asignados a una cuenta de usuario especial en todas las bases de datos llamadas **dbo**. Cualquier objeto que cree un administrador del sistema pertenece automáticamente a **dbo**. El usuario **dbo** es una cuenta predeterminada y no se puede eliminar.

Cuenta de usuario guest

La cuenta de usuario **guest** permite inicios de sesión sin cuentas de usuario para tener acceso a una base de datos. Las cuentas de inicio de sesión asumen la identidad del usuario **guest** cuando se cumplen las dos condiciones siguientes:

- La cuenta de inicio de sesión tiene acceso a SQL Server pero no tiene acceso a la base de datos a través de su propia cuenta de usuario.
- La base de datos contiene una cuenta de usuario **guest**.

Se pueden aplicar permisos al usuario **guest** como si se tratara de cualquier otra cuenta de usuario. Puede eliminar y agregar el usuario **guest** a cualquier base de datos excepto a las bases de datos **master** y **tempdb**.

Asignación de cuentas de inicio de sesión a funciones

Objetivo del tema
Presentar las funciones de SQL Server.

Explicación previa
Las funciones proporcionan un medio para agrupar usuarios en una sola unidad a la que se pueden aplicar permisos.

- Funciones fijas de servidor
- Funciones fijas de base de datos
- Funciones de base de datos definidas por el usuario
- Asignación de cuentas de inicio de sesión a cuentas de usuario y funciones

Las funciones proporcionan un medio para agrupar usuarios en una sola unidad a la que se pueden aplicar permisos.

SQL Server proporciona funciones de servidor y de base de datos predefinidas para las funciones administrativas comunes, de forma que pueda conceder fácilmente una selección de permisos administrativos a un usuario específico.

También puede crear sus propias funciones de base de datos para representar las tareas que realiza un tipo de empleados de la organización. Cuando un empleado cambia de puesto de trabajo, sólo tiene que agregar al empleado como miembro de la función; quite el empleado de la función cuando el empleado cambie de puesto de trabajo. No es necesario conceder y revocar permisos de forma repetitiva cuando los empleados empiezan o terminan una labor concreta. Si la función de un puesto de trabajo cambia, es muy sencillo modificar los permisos de la función y hacer que los cambios se apliquen automáticamente a todos los miembros de la función.

Funciones fijas del servidor

Objetivo del tema
Presentar las funciones fijas del servidor.

Explicación previa
Las funciones fijas del servidor permiten agrupar los privilegios administrativos en el nivel del servidor.

Función	Permiso
sysadmin	**Realizar cualquier actividad**
dbcreator	**Crear y alterar bases de datos**
diskadmin	**Administrar archivos de disco**
processadmin	**Administrar procesos de SQL Server**
serveradmin	**Configurar el servidor**
setupadmin	**Instalar duplicación**
securityadmin	**Administrar y auditar inicios de sesión**
bulkadmin	**Ejecutar instrucciones BULK INSERT**

Sugerencia
Destaque que la función sysadmin es el equivalente de la cuenta de inicio de sesión sa.

Los miembros de esta función pueden ejecutar cualquier actividad en el servidor.

Las funciones fijas del servidor permiten agrupar los privilegios administrativos en el nivel del servidor. Se administran de forma independiente de las bases de datos de usuario en el nivel del servidor y se almacenan en la tabla del sistema **master.sysxlogins**.

Asignación de una cuenta de inicio de sesión a una función fija de servidor

Para agregar una cuenta de inicio de sesión como miembro de una función fija del servidor, puede utilizar **Propiedades de inicio de sesión** en el Administrador corporativo de SQL Server o el procedimiento almacenado del sistema **sp_addsrvrolemember**. Sólo los miembros de las funciones fijas del servidor pueden ejecutar el procedimiento almacenado del sistema **sp_addsrvrolemember**.

Cuando se agrega una cuenta de inicio de sesión a una función del servidor, la fila correspondiente de la cuenta de inicio de sesión en la tabla **sysxlogins** se actualiza para indicar que la cuenta de inicio de sesión es miembro de la función y tiene permisos asociados con la función del servidor.

Al asignar cuentas de inicio de sesión a funciones fijas del servidor, tenga en cuenta los siguientes hechos:

- No puede agregar, modificar ni quitar funciones fijas de servidor.
- Cualquier miembro de una función fija del servidor puede agregar otras cuentas de inicio de sesión a esa función.
- El procedimiento almacenado del sistema **sp_addsrvrolemember** no se puede ejecutar en una transacción definida por el usuario.

También puede utilizar el procedimiento almacenado del sistema **sp_dropsrvrolemember** para quitar un miembro de una función fija del servidor.

Funciones fijas de base de datos

Objetivo del tema
Presentar las funciones fijas de base de datos.

Explicación previa
Las funciones fijas de base de datos permiten agrupar los privilegios administrativos en el nivel de base de datos.

Función	Permiso
public	Mantener los permisos predeterminados
db_owner	Realizar cualquier actividad de funciones
db_accessadmin	Agregar o eliminar usuarios de bases de datos, grupos y funciones
db_ddladmin	Agregar, modificar o eliminar objetos
db_security admin	Asignar permisos de funciones y objetos
db_backupoperator	Realizar copias de seguridad y restauraciones
db_datareader	Leer datos de cualquier tabla
db_datawriter	Agregar, cambiar o eliminar datos de las tablas
db_denydatareader	No puede leer los datos de cualquier tabla
db_denydatawriter	No puede cambiar los datos de cualquier tabla

Sugerencia
Destaque que la función public de la base de datos Northwind ha recibido todos los permisos. La seguridad se establece de este modo sólo porque Northwind es una base de datos de ejemplo.

Las funciones fijas de base de datos permiten agrupar los privilegios administrativos en el nivel de base de datos. Las funciones fijas de base de datos están almacenadas en la tabla del sistema **sysusers** de cada base de datos.

La función public

La función **public** es una función de base de datos especial a la que pertenecen todos los usuarios de la base de datos y que no se puede quitar. La función **public**:

- Mantiene todos los permisos predeterminados de los usuarios de una base de datos.
- No puede tener usuarios, grupos o funciones asignados, ya que los usuarios, grupos y funciones ya pertenecen a ella de forma predeterminada.
- Se encuentra en todas las bases de datos, incluidas las bases de datos **master**, **msdb**, **tempdb**, **model** y todas las bases de datos de usuario.
- No se puede eliminar.

Sin los permisos apropiados, un usuario se puede conectar a SQL Server, pero sólo puede llevar a cabo unas tareas limitadas. Sin permisos, los usuarios poseen todos los permisos que se hayan concedido a la función **public** y pueden:

- Ejecutar instrucciones que no requieran permisos, como la instrucción PRINT.
- Ver información de las tablas del sistema y ejecutar ciertos procedimientos almacenados del sistema para obtener información de la base de datos **master** y las bases de datos de usuario a las que tengan acceso.
- Tener acceso a cualquier base de datos con una cuenta **guest**.

Asignación de una cuenta de seguridad a una función fija de base de datos

Para agregar una cuenta de seguridad como miembro de una función fija de base de datos, utilice el Administrador corporativo de SQL Server o el procedimiento almacenado del sistema **sp_addrolemember**. Sólo los miembros de la función **db_owner** pueden ejecutar el procedimiento almacenado del sistema **sp_addrolemember** para cualquier función de la base de datos.

Al asignar cuentas de seguridad a una función fija de base de datos, tenga en cuenta los siguientes hechos:

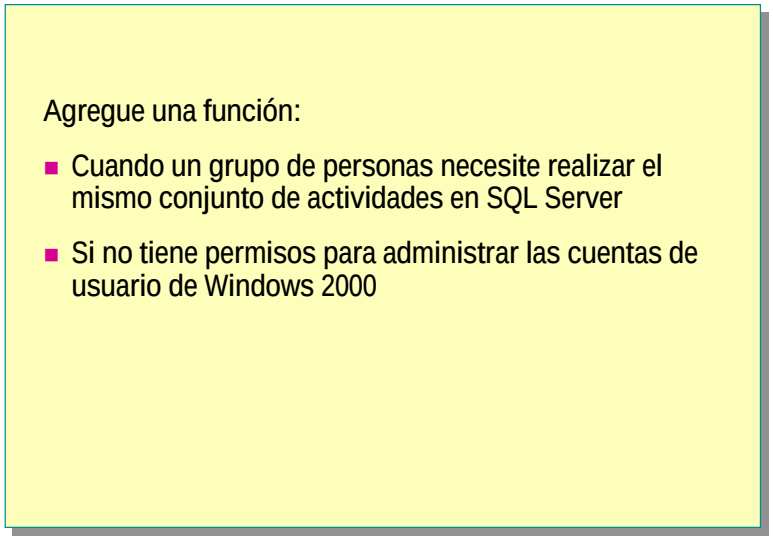
- Las funciones fijas de base de datos no se pueden agregar, modificar ni quitar.
- Cualquier miembro de una función fija de base de datos puede agregar otras cuentas de inicio de sesión a esa función.

También puede utilizar el procedimiento almacenado del sistema **sp_droprolemember** para eliminar una cuenta de seguridad de una función.

Funciones de base de datos definidas por el usuario

Objetivo del tema
Presentar las funciones de base de datos definidas por el usuario.

Explicación previa
La creación de una función de base de datos definida por el usuario permite crear un grupo de usuarios con un conjunto de permisos comunes.



La creación de una función de base de datos definida por el usuario permite crear un grupo de usuarios con un conjunto de permisos comunes. Debe agregar una función definida por el usuario a la base de datos en los casos siguientes:

- Cuando un grupo de personas necesite realizar un conjunto de actividades especificado en SQL Server y no exista un grupo de Windows 2000 aplicable.
- Si no tiene permisos para administrar las cuentas de usuario de Windows 2000.

Por ejemplo, una compañía puede constituir un comité benéfico que incluya empleados de diferentes departamentos en distintos niveles. Los empleados necesitan tener acceso a una tabla especial del proyecto en la base de datos. No existe ningún grupo de Windows 2000 que incluya sólo a esos empleados y no hay ninguna otra razón para crear uno en Windows 2000. Podría crear una función definida por el usuario, **CharityEvent**, para ese proyecto y luego agregar cuentas de usuario de Windows 2000 individuales a la función. Cuando se apliquen los permisos, las cuentas de usuario individuales de la función obtendrán acceso a la tabla.

Creación de una función de base de datos definida por el usuario

Para crear una nueva función de base de datos, utilice el Administrador corporativo de SQL Server o el procedimiento almacenado del sistema **sp_addrole**. Por cada función definida por el usuario se agrega una entrada a la tabla **sysusers** de la base de datos actual. Sólo los miembros de la función **db_securityadmin** o **db_owner** pueden ejecutar **sp_addrole**.

Al crear una función de base de datos, tenga en cuenta lo siguiente:

- Cuando aplique permisos a la función, todos los miembros de la función obtendrán el efecto del permiso, igual que si el permiso se hubiera aplicado directamente a las propias cuentas de los miembros.

Asignación de una cuenta de seguridad a una función de base de datos definida por el usuario

Después de crear una función, utilice el Administrador corporativo de SQL Server o el procedimiento almacenado del sistema **sp_addrolemember** para agregar usuarios o funciones como miembros de la función. Sólo los miembros de la función fija del servidor **sysadmin** o de las funciones fijas de base de datos **db_securityadmin** y **db_owner**, o el propietario de la función pueden ejecutar **sp_addrolemember** para agregar un miembro a una función de base de datos definida por el usuario.

Al asignar cuentas de seguridad a una función de base de datos definida por el usuario, tenga en cuenta los siguientes hechos:

- Cuando agregue una cuenta de seguridad a una función, todos los permisos aplicados a la función se aplican al nuevo miembro.
- Cuando agrega una función de SQL Server como miembro de otra función de SQL Server, no se pueden crear funciones recursivas. Por tanto, **cuentaDeSeguridad** no se podría agregar como miembro de **función** si **función** ya fuera miembro de **cuentaDeSeguridad**.

En la tabla siguiente se enumeran procedimientos almacenados del sistema adicionales que se pueden utilizar para administrar funciones de base de datos.

Procedimiento almacenado de sistema	Descripción
sp_droprole	Quita una función de SQL Server de la base de datos actual.
sp_droprolemember	Quita una cuenta de seguridad de una función de SQL Server.

Asignación de permisos a usuarios y funciones

Objetivo del tema
Presentar el concepto general de permisos.

Explicación previa
Después de haber establecido las cuentas de seguridad en una base de datos, debe asignar permisos para exigir la seguridad de la base de datos.

- Tipos de permisos
- Concesión, denegación y revocación de permisos
  - Concesión de permisos para permitir el acceso
  - Denegación de permisos para impedir el acceso
  - Revocación de permisos concedidos y denegados

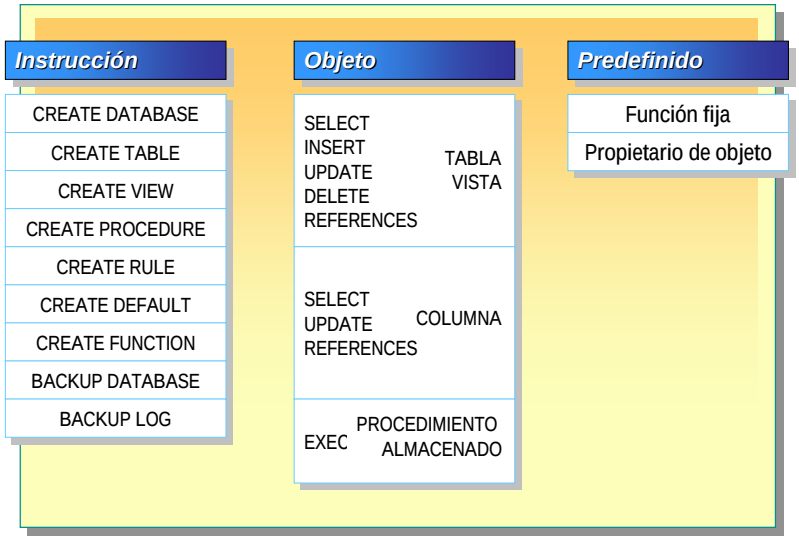
Después de haber asignado cuentas de inicio de sesión a cuentas de usuario y funciones, debe asignar permisos para exigir la seguridad de la base de datos.

Los permisos especifican para qué objetos de la base de datos tienen autorización los usuarios y qué pueden hacer los usuarios con esos objetos. Los permisos que un usuario tiene en una base de datos dependen de los permisos de la cuenta de usuario y de las funciones a las que pertenezca el usuario. Es importante diseñar los permisos que vaya a conceder a cada usuario o cada grupo. Todas las bases de datos tienen su propio sistema de permisos independiente.

Tipos de permisos

Objetivo del tema
Comparar los permisos de objeto, instrucción y los predefinidos.

Explicación previa
Hay tres tipos de permisos en SQL Server: de instrucción, de objeto y predefinidos.



Hay tres tipos de permisos en SQL Server: de instrucción, de objeto y predefinidos.

Permisos de instrucción

Las actividades relativas a la creación de bases de datos o de elementos en una base de datos requieren una clase de permisos llamada permisos de instrucción. Los permisos de instrucción ofrecen a los usuarios el privilegio de emitir ciertas instrucciones Transact-SQL. Los permisos de instrucción, como CREATE DATABASE, se aplican a la misma instrucción, no al elemento específico que se define en la base de datos. Sólo los miembros de la función **sysadmin**, **db_owner** o **db_securityadmin** pueden conceder permisos de instrucción.

Permisos de objeto

Las actividades relacionadas con los datos o la ejecución de procedimientos requieren una clase de permisos conocida como permisos de objeto.

Permisos de tablas y vistas

Los permisos de objeto para tablas y vistas controlan la capacidad de que los usuarios ejecuten las instrucciones SELECT, INSERT, UPDATE y DELETE en la tabla o la vista.

Nota Si quiere que los usuarios puedan utilizar cláusulas WHERE en instrucciones UPDATE, debe concederles la posibilidad de ejecutar instrucciones SELECT además de las instrucciones UPDATE.

Permisos de columna

Los permisos SELECT, UPDATE y REFERENCES se pueden aplicar de forma selectiva a columnas individuales.

Cuando un usuario agrega una fila a una tabla que tiene una restricción FOREIGN KEY, o cambia los datos de una columna que tiene una restricción FOREIGN KEY, SQL Server tiene que validar los datos de la columna con los datos a los que se haga referencia en la restricción FOREIGN KEY. Si el usuario no tiene permisos SELECT en la columna o la tabla de referencia, se le debe conceder el permiso REFERENCES para la columna.

Permisos de procedimientos almacenados

El permiso EXECUTE es el único permiso de objeto para los procedimientos almacenados.

Permisos predefinidos

Sólo los miembros de funciones fijas o los propietarios de los objetos de base de datos pueden desempeñar ciertas actividades. Los permisos que ejecutan dichas actividades se conocen como permisos predefinidos o implícitos.

Permisos de función fija

Las funciones fijas tienen permisos administrativos implícitos. Por ejemplo, un usuario que sea a miembro de la función **sysadmin** hereda automáticamente todos los permisos para hacer o leer cualquier cosa en una instalación de SQL Server. La función **sysadmin** tiene permisos que no se pueden cambiar, así como permisos implícitos que no se pueden aplicar a otras cuentas de usuario, como la posibilidad de configurar la instalación de SQL Server.

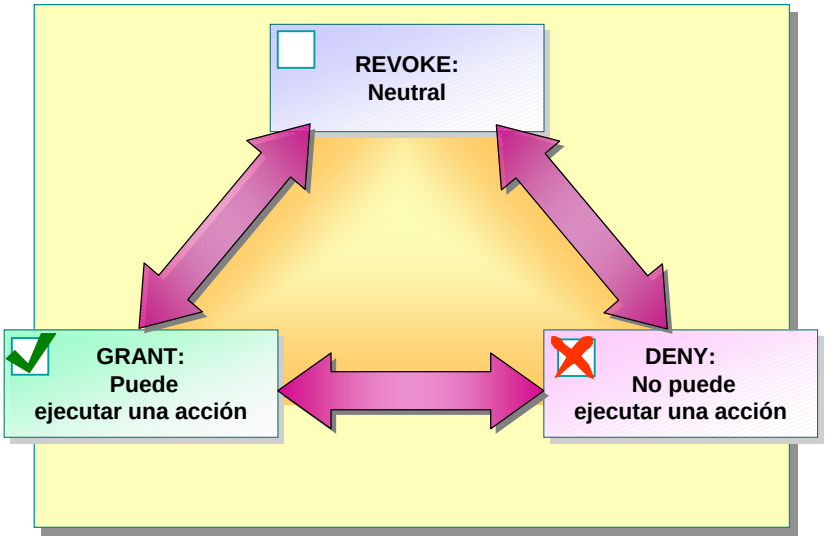
Permisos de los propietarios de objetos

Los propietarios de objetos también tienen permisos implícitos que les permiten realizar todas las actividades con los objetos de su propiedad. Por ejemplo, un usuario que sea el propietario de una tabla o un miembro de un grupo que se haya definido como propietario de la tabla, puede desempeñar cualquier actividad relativa a la tabla. El usuario puede ver, agregar o eliminar datos; alterar la definición de la tabla y controlar los permisos que permiten que otros usuarios trabajen con ella.

◆ Concesión, denegación y revocación de permisos

Objetivo del tema
Presentar los conceptos de concesión, denegación y revocación de permisos.

Explicación previa
Los permisos de un usuario o una función se pueden conceder, denegar o revocar.



Sugerencia
Destaque que la instrucción DENY impide que los usuarios realicen acciones. Prevalece sobre cualquier permiso, aunque el permiso se haya concedido directamente al usuario o a una función a la que pertenece el usuario.

Los permisos de un usuario o una función pueden estar en uno de tres estados: concedido, denegado o revocado. Los permisos que no se han concedido ni denegado a un usuario son neutros, como si se hubieran revocado. Los permisos se almacenan como entradas de la tabla del sistema **sysprotects** en cada base de datos. En la tabla siguiente se describen los tres estados de un permiso.

Instrucción	Estado de la entrada en la tabla sysprotects	Descripción
GRANT	Positivo	Puede ejecutar una acción.
DENY	Negativo	No puede ejecutar una acción y no puede ser sobrescrito por la pertenencia a una función.
REVOKE	Ninguno	No puede ejecutar una acción, pero puede ser sobrescrito por la pertenencia a una función.

Los permisos concedidos son acumulativos; los usuarios pueden ejecutar todas las acciones para las que se les haya concedido permiso individualmente y todas las acciones para las que se haya concedido permiso a las funciones a las que pertenezcan.

La instrucción DENY impide que los usuarios realicen acciones. Sobrescribe un permiso de una función a la que el usuario pertenece, tanto si se ha concedido el permiso al usuario directa como indirectamente.

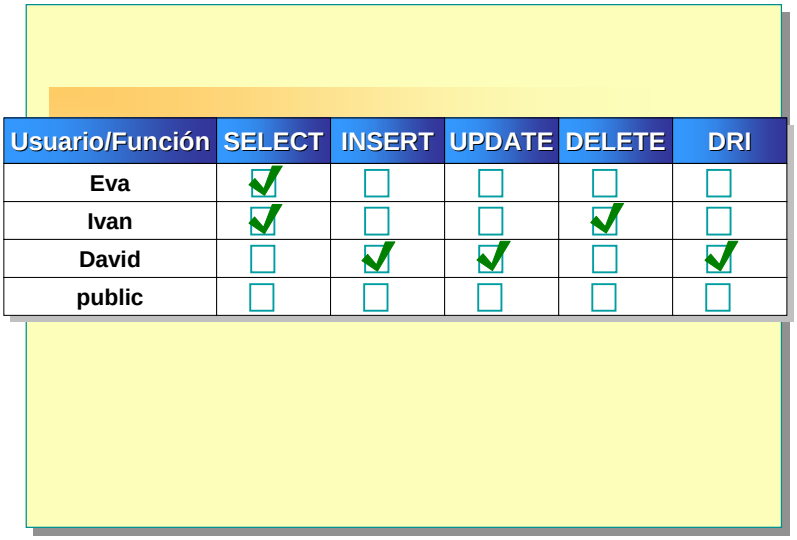
Los usuarios tienen permiso para ejecutar una acción si se dan las dos condiciones siguientes:

- Se les ha concedido el permiso directamente o pertenecen a una función a la que se ha concedido el permiso.
- El permiso no se ha denegado directamente al usuario o a una de las funciones a las que el usuario pertenece.

Concesión de permisos para permitir el acceso

Objetivo del tema
Describir cómo se conceden los permisos.

Explicación previa
Puede conceder permisos a cuentas de seguridad y permitir que las cuentas realicen actividades o trabajen con los datos de la base de datos.



Sugerencia
Muestre cómo conceder permisos de tabla, vista, procedimiento almacenado y columna en el Administrador corporativo de SQL Server.

Puede conceder permisos a cuentas de seguridad y permitir que las cuentas realicen actividades o trabajen con los datos de una base de datos.

Al conceder permisos, tenga en cuenta los siguientes hechos:

- Sólo puede conceder permisos en la base de datos actual.
- El derecho para conceder permisos corresponde de forma predeterminada a los miembros de las funciones **sysadmin**, **db_owner** y **db_securityadmin**, y a los propietarios de objetos.

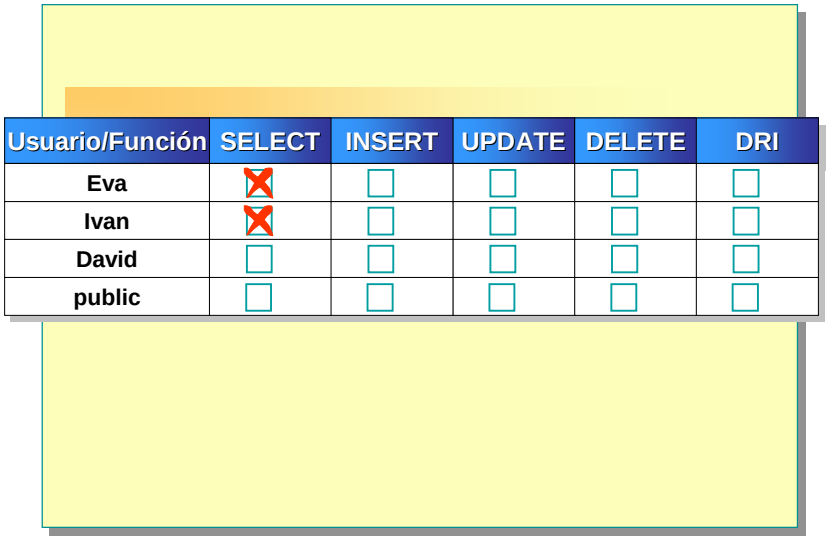
Puede conceder permisos con el Administrador corporativo de SQL Server o con la instrucción GRANT.

Los permisos disponibles varían en función de los objetos seleccionados. Por ejemplo, un procedimiento almacenado tiene permisos EXECUTE; una tabla o una vista tienen los permisos SELECT, INSERT, UPDATE, DELETE y permisos de integridad referencial; y las columnas individuales de una tabla o vista tienen los permisos SELECT y UPDATE.

Denegación de permisos para impedir el acceso

Objetivo del tema
Describir cómo se deniegan los permisos.

Explicación previa
Puede que, ocasionalmente, desee limitar los permisos de un usuario o una función; para ello, puede denegar los permisos a esa cuenta de seguridad.



Usuario/Función	SELECT	INSERT	UPDATE	DELETE	DRI
Eva	✗	☐	☐	☐	☐
Ivan	✗	☐	☐	☐	☐
David	☐	☐	☐	☐	☐
public	☐	☐	☐	☐	☐

Sugerencia
Muestre cómo denegar permisos de tabla, vista, procedimiento almacenado y columna en el Administrador corporativo de SQL Server.

Puede que, ocasionalmente, desee limitar los permisos de un usuario o una función; para ello, puede denegar los permisos a esa cuenta de seguridad. La denegación de permisos de una cuenta de seguridad:

- Quita los permisos previamente concedidos al usuario o función.
- Desactiva los permisos heredados de otra función.
- Asegura que el usuario o función no herede permisos de otra función en el futuro.

Al denegar permisos, tenga en cuenta los siguientes hechos:

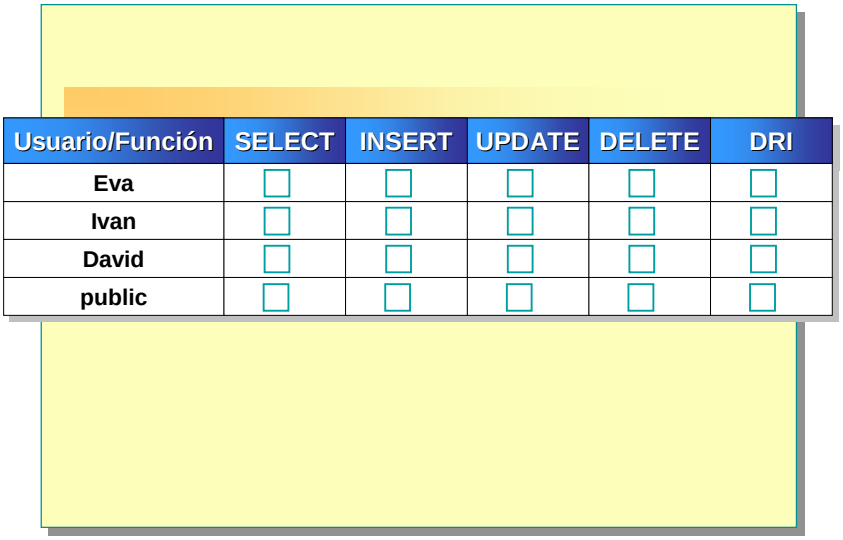
- Sólo puede denegar permisos en la base de datos actual.
- El permiso para denegar permisos corresponde de forma predeterminada a los miembros de las funciones **sysadmin**, **db_owner** y **db_securityadmin**, y a los propietarios de los objetos.

Para denegar permisos, puede utilizar el Administrador corporativo de SQL Server o la instrucción DENY.

Revocación de permisos concedidos y denegados

Objetivo del tema
Describir cómo se revocan los permisos.

Explicación previa
Para desactivar un permiso concedido o denegado, puede revocarlo.



Usuario/Función	SELECT	INSERT	UPDATE	DELETE	DRI
Eva	☐	☐	☐	☐	☐
Ivan	☐	☐	☐	☐	☐
David	☐	☐	☐	☐	☐
public	☐	☐	☐	☐	☐

Sugerencia
Muestre cómo revocar permisos de tabla, vista, procedimiento almacenado y columna en el Administrador corporativo de SQL Server.

Para desactivar un permiso concedido o denegado, puede revocarlo. La revocación de permisos es similar a la denegación de permisos en que ambas acciones quitan un permiso concedido. La diferencia estriba en que mientras la revocación de un permiso quita un permiso concedido, no impide que el usuario o la función hereden ese permiso en el futuro.

También puede quitar un permiso previamente denegado si revoca la instrucción DENY del permiso.

Al revocar permisos, tenga en cuenta los siguientes hechos:

- Sólo puede revocar permisos en la base de datos actual.
- La revocación de un permiso quita las entradas de la tabla del sistema **syspermissions** que se crearon al conceder o denegar el permiso.
- Los permisos para revocar permisos corresponden de forma predeterminada a los miembros de las funciones **sysadmin**, **db_owner** y **db_securityadmin**, y a los propietarios de los objetos.

Para quitar un permiso previamente concedido o denegado, puede utilizar el Administrador corporativo de SQL Server o la instrucción REVOKE.

Administración de la seguridad en SQL Server

Objetivo del tema
Presentar los temas que se deben tener en cuenta al administrar la seguridad.

Explicación previa
Al diseñar la seguridad, debe tener en cuenta las recomendaciones siguientes.

- Determinación del uso de cuentas de inicio de sesión predeterminadas
  - sa
  - BUILTIN\Administradores
- Determinación de la función de la cuenta de usuario guest
- Determinación de los permisos de la función public
- Aplicación de permisos a las funciones
- Creación de objetos con el propietario dbo
- Protección de los pasos de trabajo CmdExec y ActiveScripting

---

Al diseñar la seguridad, debe tener en cuenta las recomendaciones siguientes.

Para su información
Las cuentas que los alumnos utilizan durante las prácticas son miembros de la función sysadmin.

Determinación del uso de cuentas de inicio de sesión predeterminadas

Determine si va a utilizar las cuentas de inicio de sesión predeterminadas y, en caso afirmativo, cómo las va a usar.

Cuenta de inicio de sesión sa

El administrador del sistema (**sa**) es un inicio de sesión especial proporcionado para la compatibilidad con versiones anteriores. De manera predeterminada, se asigna a la función fija de servidor **sysadmin** y no se puede quitar. Aunque **sa** es una cuenta integrada de inicio de sesión de administrador, no se debe utilizar habitualmente.

Sugerencia
Otro método de restringir la cuenta de inicio de sesión BUILTIN\Administradores consiste en quitar los administradores de dominio del grupo local de Windows 2000 Administradores.

Cuenta de inicio de sesión BUILTIN\Administradores

Los administradores del sistema deben ser miembros de la función fija de servidor **sysadmin**. El grupo local de Windows 2000 **Administradores** se asigna automáticamente a la cuenta de inicio de sesión de SQL Server **BUILTIN\Administradores**, que es miembro de la función **sysadmin**.

Si no desea que todos los administradores de Windows 2000 de la organización tengan acceso completo a SQL Server, puede quitar la cuenta de inicio de sesión **BUILTIN\Administradores** o quitar la cuenta de inicio de sesión de la función **sysadmin**. Puede sustituir la cuenta de inicio de sesión por un usuario más apropiado y asignar la nueva cuenta a la función **sysadmin**. Si sustituye la cuenta de inicio de sesión por un grupo de Windows 2000, cualquier administrador de Windows 2000 podrá agregar su nueva cuenta de usuario a dicho grupo de Windows 2000.

Determinación de la función de la cuenta de usuario guest

La cuenta de usuario **guest** permite que una cuenta de inicio de sesión sin una cuenta de usuario tenga acceso a una base de datos. Debe decidir si sus bases de datos van a tener una cuenta **guest** y, si es así, qué permisos va a tener la cuenta **guest** en las bases de datos.

Determinación de los permisos de la función public

La función **public** es una función de base de datos especial a la que pertenecen todos los usuarios de la base de datos. Controla los permisos que todos los usuarios tienen de forma predeterminada en cada base de datos. Debe decidir los permisos que tendrá la función **public** en cada base de datos; de forma predeterminada, la función **public** no tiene permisos.

Aplicación de permisos a las funciones

Al aplicar permisos a funciones, siga estas directrices:

- Cree funciones definidas por el usuario para la aplicación. Por ejemplo, puede tener una función para el departamento de nóminas, otra para el departamento de recursos humanos y otra para el departamento de ventas.
- Aplique permisos a las funciones definidas por el usuario que cree para la aplicación. Para simplificar la administración de los permisos, debe evitar aplicar permisos directamente a un usuario individual.
- Agregue miembros a las funciones definidas por el usuario que cree. Utilice grupos de Windows 2000 siempre que sea posible y, después, agregue funciones o usuarios individuales.

Creación de objetos con el propietario dbo

Es muy importante determinar qué usuarios y funciones pueden crear objetos en una base de datos. En general, se recomienda que sólo utilice las funciones fijas de base de datos **sysadmin**, **db_owner** y **db_ddladmin** para crear objetos de base de datos.

Se recomienda encarecidamente que todos los objetos se definan con el usuario **dbo** como propietario de los objetos. La definición de objetos con **dbo** como propietario permite que cualquier usuario de la base de datos haga referencia al objeto sin incluir el nombre del propietario. Todos los objetos que se crean con la función **sysadmin** tienen **dbo** como propietario. En cualquier otra función, especifique siempre el usuario **dbo** como nombre del propietario cuando cree objetos; de lo contrario, el objeto se creará con su nombre de usuario como propietario del objeto.

Cambio de los propietarios de los objetos

Si los objetos no se crearon con el usuario **dbo** como propietario, puede cambiar el propietario del objeto mediante el procedimiento almacenado del sistema **sp_changeobjectowner**.

Sintaxis

sp_changeobjectowner *objeto*, *propietario*

Al cambiar los propietarios de los objetos de una base de datos, tenga en cuenta los siguientes hechos:

- Sólo los miembros de las funciones fijas de base de datos **db_owner** y **db_ddladmin**, y los miembros de la función del servidor **securityadmin** pueden cambiar los propietarios de los objetos de una base de datos.
- Cambiar el propietario de un objeto de base de datos requiere actualizar directamente todas las secuencias de comandos y los archivos por lotes que hagan referencia a ese objeto para incluir la nueva información de propiedad. SQL Server no puede realizar esta actualización automáticamente.

Protección de los pasos de trabajo CmdExec y ActiveScripting

Los trabajos de SQL Server pueden incluir los pasos CmdExec y ActiveScripting. Éstos permiten que el trabajo recorra los pasos que ejecutan las secuencias de comandos, las aplicaciones compiladas y los archivos por lotes. Cuando un usuario que no es miembro de la función **sysadmin** realiza estos trabajos, éstos se ejecutan en el contexto de la cuenta de usuario de Windows 2000 asociada al servicio Agente SQL Server. Esto puede dar por resultado que los usuarios tengan la capacidad de ejecutar funciones en el entorno Windows en el que se han denegado sus cuentas de Windows.

Para impedir que esto suceda, puede configurar el servicio Agente SQL Server para que permita que sólo los usuarios con privilegios **sysadmin** ejecuten pasos de trabajo mediante CmdExec y ActiveScripting.

Administración de la seguridad de la aplicación

Objetivo del tema
Presentar el tema de la administración de la seguridad de la aplicación.

Explicación previa
Puede proteger el acceso a una base de datos mediante la autenticación de la cuenta de inicio de sesión y los permisos. También puede utilizar métodos para proteger el acceso en el nivel de aplicación, como las vistas, procedimientos almacenados y funciones de aplicación.

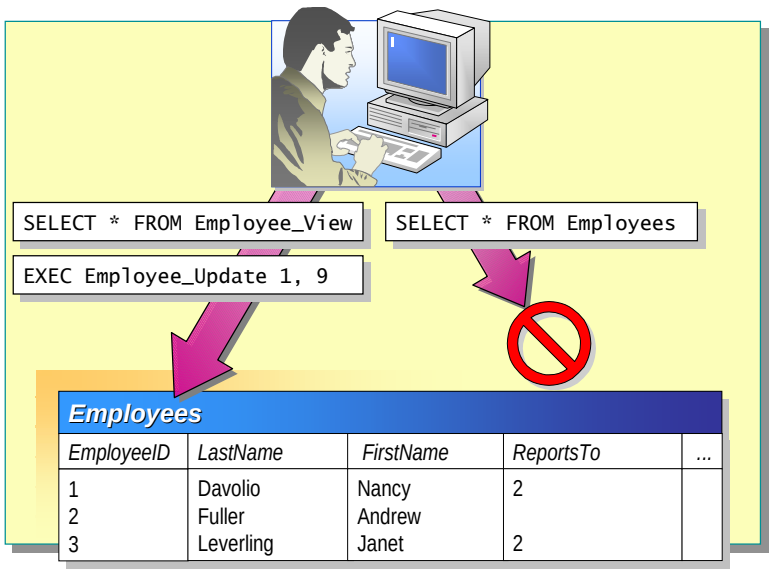
- Administración de la seguridad con vistas y procedimientos almacenados
- Administración de la seguridad de las aplicaciones de cliente con funciones de aplicación

Puede proteger el acceso a una base de datos mediante la autenticación de la cuenta de inicio de sesión y los permisos. También puede utilizar métodos para proteger el acceso en el nivel de aplicación, como las vistas, procedimientos almacenados y funciones de aplicación.

Administración de la seguridad con vistas y procedimientos almacenados

Objetivo del tema
Describir cómo se utilizan las vistas y los procedimientos almacenados para simplificar los permisos.

Explicación previa
Las vistas y los procedimientos almacenados proporcionan un método secundario para ofrecer a los usuarios acceso a los datos y la posibilidad de realizar actividades en una base de datos.



Sugerencia
Destaque que las vistas se pueden utilizar para impedir que los usuarios sepan que en una tabla hay otras columnas a las que no tienen acceso.

Las vistas y los procedimientos almacenados proporcionan un método secundario para ofrecer a los usuarios acceso a los datos y la posibilidad de realizar actividades en una base de datos. Le permiten configurar la seguridad con los objetos de SQL Server que se crean para una aplicación.

Las vistas y los procedimientos almacenados le permiten administrar los permisos sólo sobre la vista o el procedimiento almacenado, en lugar de los permisos sobre los objetos a los que hacen referencia. También protegen a los usuarios de los cambios efectuados en las tablas subyacentes.

Si desea utilizar vistas y procedimientos almacenados para simplificar la seguridad:

- Conceda permisos sobre una vista a los usuarios sin conceder permisos sobre las tablas subyacentes.

  Por ejemplo, la columna **BirthDate** de una tabla contiene información confidencial de los empleados, pero las demás columnas contienen información a la que los usuarios deben tener acceso. Puede definir una vista que incluya todas las columnas de la tabla excepto la columna **BirthDate**. Mientras la tabla y la vista tengan el mismo propietario, al conceder permisos SELECT sobre la vista se permite que los usuarios vean las columnas no confidenciales sin conceder permisos sobre la tabla.
- Conceda permisos para ejecutar un procedimiento almacenado a los usuarios sin concederles acceso a las tablas que dichos procedimientos modifican.

  En un escenario de archivo, los datos anteriores a un intervalo especificado se copian en una tabla de archivo y, después, se eliminan de la tabla principal. Se pueden utilizar permisos para impedir que los usuarios eliminen directamente las filas de la tabla principal o que inserten filas en la tabla de archivo sin eliminarlas de la tabla principal. Puede crear un procedimiento almacenado que asegure que ambas actividades se ejecutan

juntas y, después, puede conceder permisos a los usuarios para que ejecuten el procedimiento almacenado.

Para crear vistas en la ventana de diseño, puede utilizar el Administrador corporativo de SQL Server. Si agrega las tablas seleccionadas al panel de diagrama y selecciona columnas para mostrar podrá designar vistas sin necesidad de aprender Transact-SQL. Otra posibilidad que tiene es la de ejecutar el Asistente para creación de vistas.

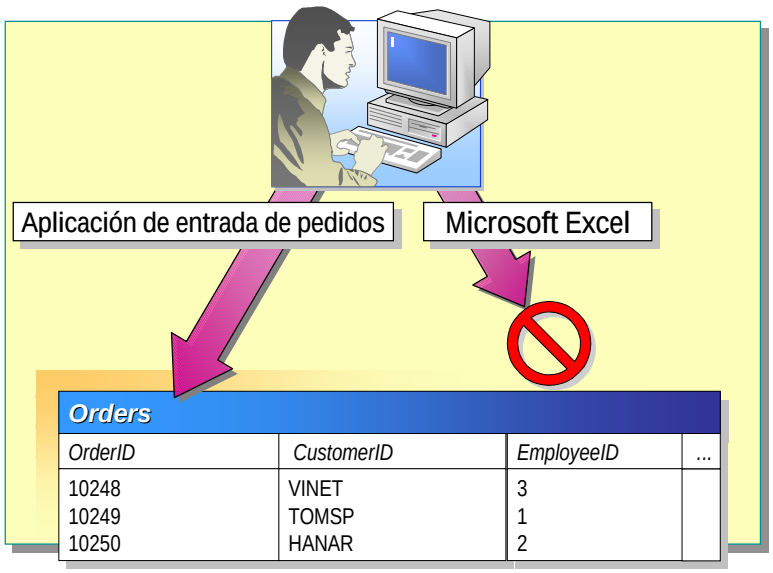
Puede crear procedimientos almacenados en el Administrador corporativo de SQL Server con el Asistente para creación de procedimientos almacenados. Para procedimientos más complejos, puede escribir las instrucciones Transact-SQL directamente en las definiciones de los procedimientos almacenados.

El Analizador de consultas SQL proporciona plantillas de secuencias de comandos de Transact-SQL para ayudarle a crear objetos en una base de datos, entre los que se incluyen tanto vistas como procedimientos almacenados. Las secuencias de comandos contienen parámetros que le ayudan a personalizar las secuencias de comandos de Transact-SQL y el cuadro de diálogo **Reemplazar parámetros de plantilla** que le guiará a lo largo del proceso.

Administración de la seguridad de las aplicaciones de cliente con funciones de aplicación

Objetivo del tema
Presentar las funciones de aplicación.

Explicación previa
Las funciones de aplicación le permiten crear un entorno de seguridad para una aplicación.



Las funciones de aplicación permiten exigir la seguridad para una aplicación determinada. Le permiten restringir que los usuarios tengan acceso indirectamente a los datos, sólo a través de una aplicación específica.

Por ejemplo, puede que desee que los empleados que reciben los pedidos sean capaces de actualizar las tablas **Products**, **Orders** y **Customer** sólo cuando utilicen la aplicación de entrada de pedidos. No desea que esos empleados tengan acceso a las tablas desde otro producto, como Microsoft Excel. En esta situación, podría crear una función de aplicación para la aplicación de entrada de pedidos.

Las funciones de aplicación son diferentes de las demás funciones. En la lista siguiente se enumeran las diferencias fundamentales entre las funciones de aplicación y las otras funciones:

- Las funciones de aplicación no tienen miembros, las activan los usuarios cuando ejecutan la aplicación.

  Esto permite que los usuarios de la aplicación tengan permisos especiales sólo cuando utilizan la aplicación y evita tener que conceder permisos a los usuarios directamente.
- Las funciones de aplicación requieren una contraseña para activarse.

Después de asumir una función de aplicación, el usuario:

- Pierde todos los permisos existentes en la base de datos actual para su cuenta de usuario y cualquier función a la que pertenezca, excepto los permisos que se apliquen a la función **public**.
- Hereda todos los permisos de la función de aplicación en la base de datos actual.

El usuario no puede ejecutar una instrucción USE DATABASE, pero puede tener acceso a otras bases de datos si utiliza nombres de objetos totalmente cualificados, siempre y cuando la base de datos tenga una cuenta **guest**. Usted debe conceder permisos a la cuenta **guest**.

Creación de funciones de aplicación

Objetivo del tema
Describir cómo se crean las funciones de aplicación.

Explicación previa
Puede crear funciones de aplicación y asignarles permisos.

- La creación de una función de aplicación inserta una fila en la tabla sysusers
- Administración de los permisos de las funciones de aplicación

Puede crear funciones de aplicación y asignarles permisos.

Creación de funciones de aplicación

Para crear una nueva función de aplicación, utilice el Administrador corporativo de SQL Server o el procedimiento almacenado del sistema **sp_addapprole**. Sólo los miembros de las funciones **db_owner**, **db_securityadmin** y **sysadmin** pueden ejecutar el procedimiento almacenado del sistema **sp_addapprole**.

Puede crear una función de aplicación de la misma manera en que crea una función de base de datos definida por el usuario, mediante el Administrador corporativo de SQL Server, siempre que seleccione **Función de aplicación** para el tipo de función de base de datos.

Al crear nuevas funciones de aplicación, tenga en cuenta los siguientes hechos:

- La creación de una función de aplicación agrega una cuenta de seguridad para la nueva función; para ello, inserta una fila a la tabla **sysusers** de la base de datos actual.
- El valor **Contraseña** es la contraseña que se requiere para activar la función.

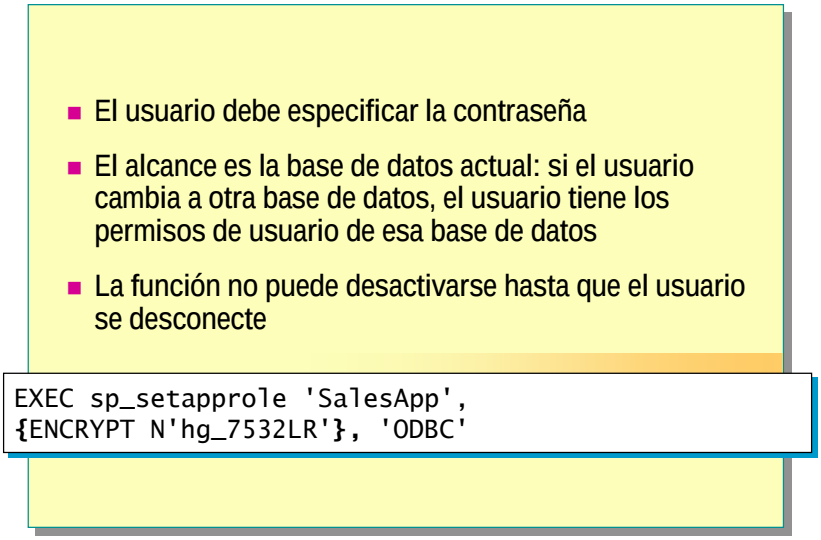
Administración de los permisos de las funciones de aplicación

Para administrar los permisos de las funciones de aplicación, utilice el Administrador corporativo de SQL Server o las instrucciones GRANT, DENY y REVOKE de Transact-SQL.

Activación de funciones de aplicación

Objetivo del tema
Describir cómo se activa una función de aplicación.

Explicación previa
Ejecute el procedimiento almacenado del sistema **sp_setapprole** para activar los permisos que están asociados con una función de aplicación.



Después de que un cliente se conecta a SQL Server con cualquier cuenta de inicio de sesión, el cliente debe ejecutar el procedimiento almacenado del sistema **sp_setapprole** para activar los permisos asociados con una función de aplicación. El procedimiento almacenado del sistema **sp_setapprole** sólo se puede ejecutar mediante instrucciones Transact-SQL directas; no se puede ejecutar desde otro procedimiento almacenado o desde una transacción definida por el usuario.

Sintaxis

sp_setapprole [@rolename =] '*función*' ,
[**@password** =] **{Encrypt N** '*contraseña*'**}** | '*contraseña*'
[,[**@encrypt** =] '*estiloDeCifrado*']

Al activar funciones de aplicación, tenga en cuenta los siguientes hechos:

- La aplicación actual debe proporcionar la contraseña, que puede estar cifrada.
- El ámbito de una función de aplicación es únicamente el de la base de datos actual. Si los usuarios cambian de base de datos, podrán desempeñar actividades en función de los permisos guest de dicha base de datos. Si no existe ninguna cuenta de usuario guest, significa que el acceso está denegado.
- Después de activar una función de aplicación con el procedimiento almacenado del sistema **sp_setapprole**, la función no se puede desactivar en la base de datos actual hasta que el usuario se desconecte de SQL Server.
- Las funciones de aplicación no se benefician del agrupamiento de conexiones de la Conectividad abierta de bases de datos (ODBC). El agrupamiento permite a la aplicación utilizar una conexión de un grupo de conexiones que no es necesario restablecer cada vez que se utiliza. Esto puede mejorar el rendimiento de la aplicación pero depende de que las propiedades de la conexión (por ejemplo, el nombre de inicio de sesión, que determina los permisos) permanezcan sin alterar.

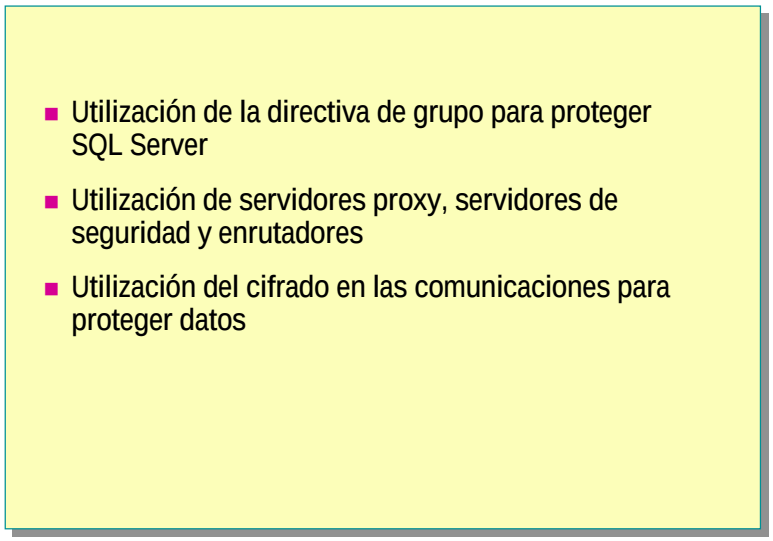
Es necesario proporcionar una contraseña para autenticar la aplicación. Opcionalmente, puede cifrar la contraseña con la función ODBC ENCRYPT. Si cifra la contraseña, ésta se debe convertir en cadena Unicode, anteponiendo a la contraseña la letra N.

También puede utilizar el procedimiento almacenado del sistema **sp_dropapprole** para quitar una función de aplicación de la base de datos actual, o el procedimiento almacenado del sistema **sp_approlepassword** para cambiar la contraseña de una función de aplicación.

Administración de la seguridad de SQL Server en la empresa

Objetivo del tema
Presentar consideraciones adicionales relacionadas con la seguridad necesarias al utilizar SQL Server en la empresa.

Explicación previa
En esta sección trataremos algunas consideraciones adicionales relacionadas con la seguridad necesarias al trabajar en una empresa.



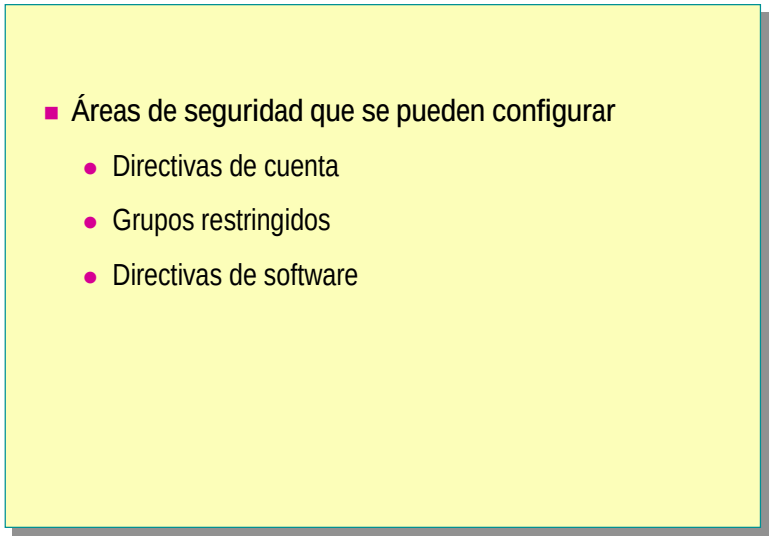
Al utilizar SQL Server en un entorno empresarial, se deben tener en cuenta ciertas cuestiones relacionadas con la seguridad. Puede utilizar Directiva de grupo para proporcionar configuraciones de seguridad a varios usuarios y equipos. Los servidores proxy, servidores de seguridad y enrutadores pueden proteger la red interna de accesos internos no autorizados, y usted podrá utilizar funciones de cifrado en las comunicaciones en Windows 2000 para garantizar que no se puedan leer los paquetes de datos en caso de que se tenga acceso a ellos.

Nota Esta sección es una introducción de alto nivel de los temas relacionados con la seguridad en un entorno empresarial.

Utilización de la directiva de grupo para proteger SQL Server

Objetivo del tema
Introducir la directiva de grupo de Windows 2000

Explicación previa
En Windows 2000 puede utilizar la directiva de grupo para definir…



En Windows 2000 puede utilizar la directiva de grupo para definir las configuraciones de usuario y equipo para grupos de usuarios y equipos. La directiva de grupo se puede utilizar para configurar muchas opciones, entre las que se incluyen directivas basadas en el Registro, opciones de seguridad, e instalación de software.

Las áreas de seguridad que se pueden configurar para los equipos incluyen:

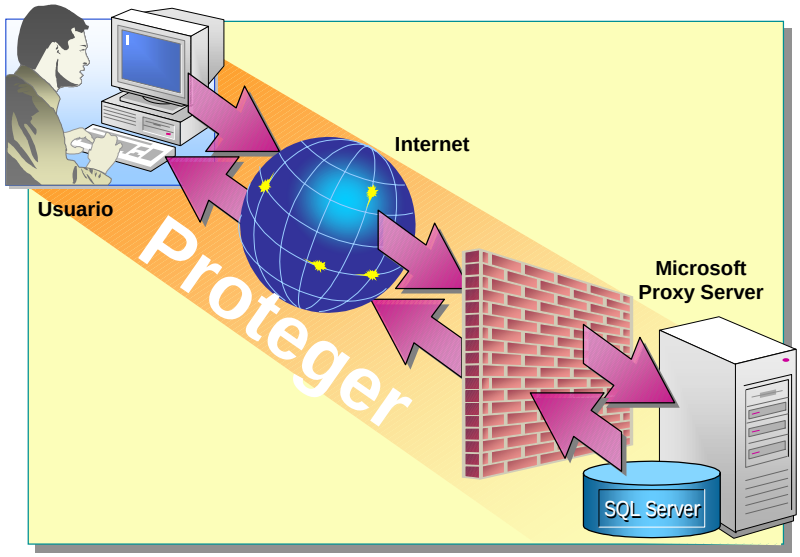
- *Directivas de cuenta*. Las directivas de cuenta son configuraciones de seguridad de equipos para la directiva de contraseña, la directiva de bloqueo y directiva Kerberos en dominios de Windows 2000.
- *Grupos restringidos*. Los grupos restringidos permiten controlar quién pertenece a un grupo determinado así como los grupos a los que debería pertenecer un grupo restringido. Los grupos restringidos permiten a los administradores exigir directivas de seguridad relacionadas con grupos importantes, como Administradores de la empresa o Nóminas.

  Por ejemplo, puede decidir que sólo José y María sean miembros del grupo Administradores de la empresa. Se pueden utilizar grupos restringidos para exigir dicha directiva. Si se agrega un tercer usuario al grupo (por ejemplo, para llevar a cabo alguna tarea en una situación de emergencia), la siguiente vez que se exija la directiva se quitará a ese tercer usuario automáticamente del grupo Administradores de la empresa.
- *Directivas de software*. Se puede utilizar la directiva de grupo para definir los valores predeterminados que se aplicarán automáticamente a los usuarios y equipos. Estos valores de configuración pueden determinar las opciones de seguridad y controlar qué software está disponible para determinados grupos de usuarios.

Utilización de servidores proxy, servidores de seguridad y enrutadores

Objetivo del tema
Describir el objeto de los servidores proxy, servidores de seguridad y enrutadores.

Explicación previa
Si conecta SQL Server a Internet se pueden originar problemas de seguridad adicionales.



Si conecta SQL Server a Internet se pueden originar problemas de seguridad adicionales. Es necesario que se asegure de que sólo los usuarios autorizados pueden tener acceso a la base de datos del servidor y que éstos sólo pueden tener acceso a los recursos necesarios para el proceso empresarial.

Conexiones a SQL Server a través de un servidor proxy

Puede permitir conexiones a SQL Server a través de Microsoft Proxy Server, una aplicación independiente que proporciona acceso seguro a y desde Internet. Esto le permite impedir que usuarios no autorizados se conecten a su red privada. Protege los datos confidenciales mediante el control de todos los permisos y del acceso al puerto escucha.

Seguridad mediante servidores de seguridad

Los servidores de seguridad constituyen una de las mejores maneras de proteger el sistema frente a ataques de usuarios a través de Internet. Un *servidor de seguridad* restringe tanto el acceso entrante como el saliente, concediendo permisos a los usuarios de Windows o impidiendo que los datos pasen a través de determinados puertos TCP/IP. También analizan todo el tráfico que se produce entre la red e Internet.

Puede utilizar las características del servidor de seguridad de Proxy Server para controlar el flujo de información que va a Proxy Server y el que viene de él. Proxy Server admite el filtrado de paquetes entrantes y salientes. Determina de forma dinámica qué paquetes pueden pasar al circuito de red interno y a los servicios proxy de nivel de aplicación. Los filtros de paquetes individuales se configuran para que sólo los paquetes especificados pasen a través de Proxy Server. Esto permite que los puertos se abran automáticamente, sólo cuando es necesario, y que se cierren en cuanto finalice la comunicación. Esta práctica reduce el número de puertos expuestos en cualquier dirección y proporciona un alto nivel de seguridad a la red.

A los sitios Web normalmente se tiene acceso a través del puerto 80 y este puerto se debe habilitar para que los usuarios de Internet puedan tener acceso a su sitio Web. Si el sitio utiliza páginas Active Server (ASP) o algún otro mecanismo de secuencias de comandos para tener acceso a los datos desde SQL Server y devolverlos al cliente como Lenguaje de marcado de hipertexto (HTML), entonces no necesitará habilitar ningún otro puerto.

Muchas aplicaciones basadas en Web utilizan dos servidores de seguridad. El primero separa el servidor Web de Internet y permite el acceso sólo a través del puerto 80. El servidor Web reside en un área semisegura denominada *red de perímetro* y tiene acceso a la base de datos de SQL Server a través de un segundo servidor de seguridad situado entre la red de perímetro y la corporativa. El número de socket oficial de la Autoridad para la asignación de números en Internet (IANA, *Internet Assigned Numbers Authority*) para SQL Server es 1433 (aunque se puede configurar estableciendo una clave de Registro). Al configurar el acceso a SQL Server mediante TCP/IP, es necesario dejar que pasen los datos a través de este puerto. A una instancia con nombre de SQL Server se le asignará automáticamente un puerto cuando se inicie la instancia, pero si desea utilizar servicios proxy con una instancia con nombre, debería sobrescribir ésta o asignar un puerto fijo a la instancia con nombre utilizando la Herramienta de red de SQL Server. Si la comunicación se efectúa a través de COM+, el puerto 135 debe estar habilitado junto con un intervalo de valores que el protocolo COM+ utiliza para negociar la conectividad con el cliente.

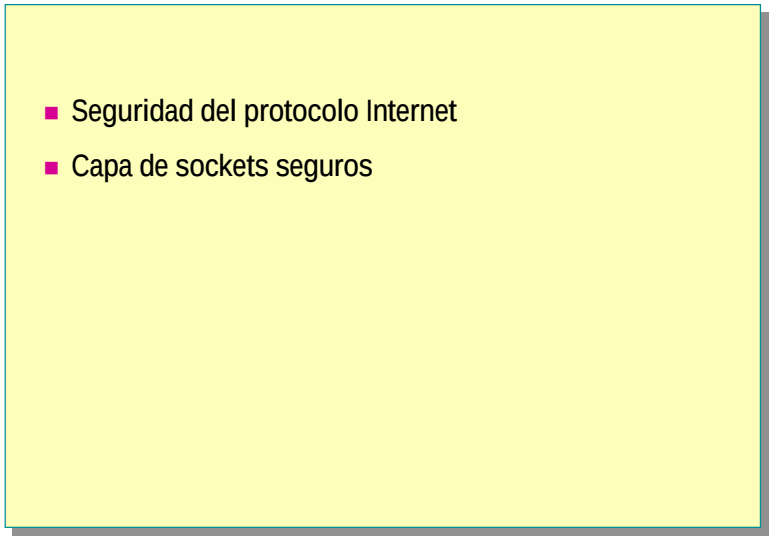
Enrutadores

Las características de enrutador incorporadas en los servidores Windows 2000 permiten a los servidores actuar como enrutadores. Un *enrutador* (o puerta de enlace) es un dispositivo que reenvía paquetes de Protocolo Internet (IP) de una red a otra. Para configurar los datos que entran en la red y que salen de ella, puede utilizar un enrutador junto con un servidor de seguridad.

Utilización del cifrado en las comunicaciones para proteger datos

Objetivo del tema
Presentar el tema del cifrado en las comunicaciones.

Explicación previa
Al transmitir paquetes de datos a través de una red o de Internet, debe asegurarse de que no se puedan leer los paquetes en caso de que se tenga acceso a ellos.



Al transmitir paquetes de datos a través de una red o de Internet, debe asegurarse de que no se puedan leer los paquetes en caso de que se tenga acceso a ellos.

Seguridad del protocolo Internet

Windows 2000 incorpora Seguridad del protocolo Internet (IPSec) para la protección de los datos a través del tráfico de la red. Es un conjunto de estándares de Internet que utiliza seguridad de cifrado para proporcionar:

- *Confidencialidad*. El tráfico IPSec está cifrado. El tráfico IPSec capturado no es inteligible sin la clave de cifrado.
- *Autenticación*. IPSec autentica al emisor de paquetes de datos IP mediante la autenticación Kerberos, certificados digitales o una clave secreta compartida.
- *Integridad de los datos*. El tráfico IPSec contiene una suma de comprobación cifrada que incorpora la clave de cifrado. El receptor puede comprobar que el paquete no ha sido modificado por el camino.

Se puede configurar la directiva de seguridad IPSec para cada dominio o para cada equipo local definiendo un conjunto de reglas y filtros que se aplicarán para regular la comunicación segura con clientes IPSec determinados.

Capa de sockets seguros

SQL Server puede utilizar el nivel de socket seguro (SSL) para cifrar todos los datos transmitidos entre un cliente y un servidor.

El cifrado SSL sólo funciona con instancias de SQL Server a las que se ha asignado un certificado de una autoridad emisora de certificados (CA) pública (CA). Los certificados de servidor permiten al cliente identificar con toda seguridad el servidor antes de compartir información confidencial. El equipo cliente debe tener también un certificado de una entidad emisora de certificados raíz de la misma autoridad. Los certificados de cliente contienen información personal acerca de los clientes que solicitan acceso al servidor.

La información de los certificados se utiliza para cifrar los datos. De esta manera se garantiza que sólo el servidor y el cliente pueden descifra la información.

Se puede utilizar la Herramienta de red de SQL Server para habilitar el cifrado SSL para todos los protocolos de servidor habilitados. El cifrado se activa o desactiva para todos los protocolos de servidor habilitados y no se puede especificar el cifrado para un protocolo concreto.

Nota Para tener compatibilidad con versiones anteriores de SQL Server, la biblioteca de red multiprotocolo continúa aceptando su propio cifrado. Este cifrado se especifica de manera independiente del cifrado SSL y no requiere la utilización de certificados. No es compatible con las instancias con nombre.
