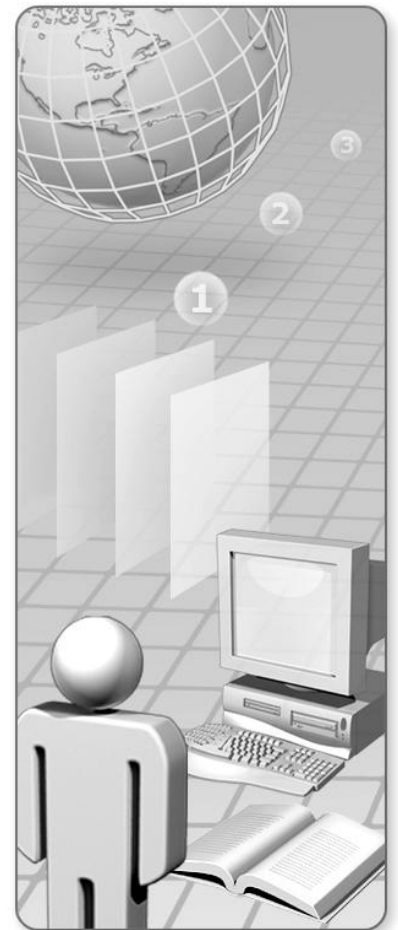


Protecting Data with Transparent Data Encryption

Table of Contents

Before You Begin	1
Exercise 1: Copying an Unencrypted Database	3
Exercise 2: Implementing Transparent Data Encryption	5
Exercise 3: Attempting to Copy an Encrypted Database	7



Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

The names of manufacturers, products, or URLs are provided for informational purposes only and Microsoft makes no representations and warranties, either expressed, implied, or statutory, regarding these manufacturers or the use of the products with any Microsoft technologies. The inclusion of a manufacturer or product does not imply endorsement of Microsoft of the manufacturer or product. Links are provided to third party sites. Such sites are not under the control of Microsoft and Microsoft is not responsible for the contents of any linked site or any link contained in a linked site, or any changes or updates to such sites. Microsoft is not responsible for webcasting or any other form of transmission received from any linked site. Microsoft is providing these links to you only as a convenience, and the inclusion of any link does not imply endorsement of Microsoft of the site or the products contained therein.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Copyright © 2007 Microsoft Corporation. All rights reserved.

Microsoft, Excel, Office, and SQL Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Before You Begin

Estimated time to complete this lab

45 minutes

Objectives

After completing this lab, you will be able to:

- Demonstrate how unencrypted data can be stolen by restoring a backup file.
- Encrypt sensitive data by using Transparent Data Encryption.
- Demonstrate that an encrypted database cannot be stolen by restoring a backup file.

Prerequisites

Before working on this lab, you must have:

- Experience of backing up and restoring Microsoft® SQL Server® databases.

Lab scenario

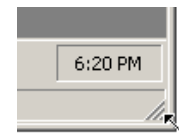
Adventure Works Cycles stores sensitive data in its databases. You regularly back up this data and store the backup files offsite. Concerns have been raised about the security of the data both onsite and offsite. You require a security solution that will protect sensitive data from the possible theft of database or backup media. However, this solution must not require significant client application redevelopment.

Virtual PC

This lab makes use of Microsoft Virtual PC 2007, which is an application that allows you to run multiple virtual computers on the same physical hardware. During the lab, you will use a virtual machine running Windows Server® 2003.

Before you start the lab, familiarize yourself with the following basics of Virtual PC:

- To switch the focus for your mouse and keyboard to the virtual machine, click inside the virtual machine window.
- To remove the focus from a virtual machine, move the mouse pointer outside the virtual machine window.
- To mimic the CTRL+ALT+DELETE key combination inside a virtual machine, use RIGHT-ALT+DELETE. In Virtual PC, the RIGHT-ALT key is called the host key.
- To enlarge the size of the virtual machine window, drag the lower-right corner of the window as seen in the screenshot.



- To switch to and from full-screen mode, press RIGHT-ALT+ENTER.

Computers in this lab

This lab uses one computer as described in the following table. Before you begin the lab, you must start the virtual machines and then log on to the computer. In each exercise, you only have to start the virtual machine that is needed.

Virtual Machine	Computer Name	User Name	Password
SQL2008CTP6HOLs	CHICAGO	Administrator	Pass@word1

Start the virtual machine

1. Launch Microsoft Virtual PC from the **Start** menu or desktop. If the Virtual PC console does not appear, double-click its icon in the notification area.
2. Select **SQL2008CTP6HOLs**, and then click **Start**.
3. When the virtual server is running, in the virtual server window, on the **Action** menu, click **Ctrl+Alt+Del** (or press RIGHT-ALT+DELETE on your keyboard) to send a CTRL+ALT+DEL sequence to the logon dialog box within the virtual server window.
4. Type the following information, and then click **OK**:
 - User name: **Administrator**
 - Password: **Pass@word1**

Exercise 1: Copying an Unencrypted Database

In this exercise, you will back up a database and restore it to another instance to verify that stolen backup files can be used to access data. Backups are essential to provide disaster recovery and archive storage. If backup media is stolen, it can be used to restore a database on a remote system and security can be compromised.

Back up the AdventureWorks database

1. Start SQL Server Management Studio.
2. In the **Connect to Server** dialog box, click **Connect** after verifying the following settings:
 - **Server type:** Database Engine
 - **Server name:** (local)\SQLDEV01
 - **Authentication:** Windows Authentication
3. In Object Explorer, expand **Databases**, right-click **AdventureWorks**, point to **Tasks**, and then click **Back Up**.
4. In the **Destination** section, click **Remove** to remove any existing backup destinations.
5. Click **Add**, type **C:\SQLHOLS\TDE\Starter\AdventureWorks.bak** and then click **OK**.
6. Click **OK** to back up the database, and then click **OK** when the backup operation has completed successfully.

Restore the AdventureWorks database on another instance of SQL Server

1. In Object Explorer, click **Connect**, and then click **Database Engine**.
2. In the **Connect to Server** dialog box, click **Connect** after verifying the following settings:
 - **Server type:** Database Engine
 - **Server name:** (local)\SQLDEV02
 - **Authentication:** Windows Authentication
3. On the (local)\SQLDEV02 connection, right-click **Databases**, and then click **Restore Database**.
4. In the **To database** box, type **AdventureWorksNonEncrypt**
5. Select **From device**, and then click the ellipses (...) button.
6. Click **Add**, navigate to **C:\SQLHOLS\TDE\Starter**, double-click **AdventureWorks.bak**, and then click **OK**.
7. In the **Restore** column, select the check box.

- In the **Select a page** list, click the **Options** page, and then select the **Overwrite the existing database** check box.
- In the **Restore As** column, change the file destinations as shown in the following table.

Original File Name	Restore As
AdventureWorks_Data	C:\SQLHOLS\TDE\Starter\AdventureWorksNonEncrypt.mdf
AdventureWorks_Log	C:\SQLHOLS\TDE\Starter\AdventureWorksNonEncrypt_log.ldf

- Click **OK** to restore the database, and then click **OK** when the restore operation has completed successfully.

Access the copied data

- Click **New Query**.
- If prompted, connect to the **(local)\SQLDEV02** database engine instance.
- Type the following code, and then click **Execute**.

```
USE AdventureWorksNonEncrypt
GO
SELECT * FROM HumanResources.Employee
```

- Confirm that the data is accessible.
- Keep SQL Server Management Studio open for the next exercise.

Key Point: Stolen backup media is a security risk because data can be restored and accessed on a remote system.

Exercise 2: Implementing Transparent Data Encryption

In this exercise, you will encrypt a database and access it from a client application to verify that encryption is transparent to client applications. Data encryption prevents the risks demonstrated in the previous exercise. Unfortunately, client applications that use traditional data encryption must decrypt the data before they can use it. This causes increased complexity for applications that use the encrypted data and slows development time. Transparent data encryption performs all encryption and decryption on the database system. This provides protection and does not require any additional work for application developers.

Configure Transparent Data Encryption

1. In SQL Server Management Studio, in Object Explorer, select the connection to the **(local)\SQLDEV01** database engine instance, and then click **New Query**. If prompted, connect to the **(local)\SQLDEV01** database server instance.
2. In the query editor window, type Transact-SQL code to create a master key with the password *Pa\$\$wOrd*.
3. Add Transact-SQL code to create a certificate with the name of *ServerCertificate* and the subject *Server level certificate*.
4. Add Transact-SQL code to create a database encryption key that uses the *AES_128* algorithm and is encrypted with the server certificate *ServerCertificate*.
5. Add Transact-SQL code to alter the **AdventureWorks** database and set encryption on.

Tip: To see the completed code, open C:\SQLHOLS\TDE\Solution\Encrypt.sql.

6. Execute the Transact-SQL query.
7. Keep SQL Server Management Studio open for the next exercise.

Access the encrypted database from a client application

1. Start Microsoft Office Excel® 2007.
2. On the **Data** tab of the ribbon, in the **Get External Data** section, click **From Other Sources**, and then click **From SQL Server**.
3. In the **Server name** box, type **(local)\SQLDEV01** and then click **Next**.
4. In the **Select the database that contains the data you want** drop-down box, select **AdventureWorks**.
5. In the **Connect to a specific table** list, click **Employee**, and then click **Next**.
6. Click **Finish**, and then click **OK** to import the data.

7. Notice that SQL Server has transparently decrypted the data without any changes to the client application.
8. Close Excel without saving changes.

Key Point: Transparent data encryption requires no configuration or programming on the client application.

Exercise 3: Attempting to Copy an Encrypted Database

In this exercise, you will back up an encrypted database and attempt to restore it to another instance to verify that stolen encrypted backup files cannot be used to access data. Backup media is often stored offsite and so it is essential that this media cannot be used to access sensitive data.

Back up the encrypted AdventureWorks database

1. In SQL Server Management Studio, in Object Explorer, in the connection to the **(local)\SQLDEV01** database engine instance, under the **Databases** folder, right-click **AdventureWorks**, point to **Tasks**, and then click **Back Up**.
2. In the **Destination** section, click **Remove** to remove any existing backup destinations.
3. Click **Add**, type **C:\SQLHOLS\TDE\Starter\AdventureWorksEncrypt.bak** and then click **OK**.
4. Click **OK** to back up the database, and then click **OK** when the backup operation has completed successfully.

Attempt to restore the encrypted backup

1. In Object Explorer, right-click the **Databases** folder for the **(local)\SQLDEV02** database engine connection, and then click **Restore Database**.
2. In the **To database** box, type **AdventureWorksEncrypt**
3. Select **From device**, and then click the ellipses (...) button.
4. Click **Add**, navigate to **C:\SQLHOLS\TDE\Starter**, double-click **AdventureWorksEncrypt.bak**, and then click **OK**.
5. In the **Restore** column, select the check box.
6. In the **Select a page** list, click the **Options** page, and then select the **Overwrite the existing database** check box.
7. In the **Restore As** column, change the file destinations as shown in the following table.

Original File Name	Restore As
AdventureWorks_Data	C:\SQLHOLS\TDE\Starter\AdventureWorksEncrypt.mdf
AdventureWorks_Log	C:\SQLHOLS\TDE\Starter\AdventureWorksEncrypt_log.ldf

8. Click **OK** to restore the database.
9. Notice that the restore operation fails because the correct server certificate cannot be found and the encrypted data is inaccessible. Then click **OK** on the error message, and cancel the restore operation.

10. Close SQL Server Management Studio. Do not save your files if prompted.
11. Close Virtual PC and discard changes.

Key Point: Although the encryption is completely transparent to client applications, the data is encrypted and backups cannot be restored to other servers without the server master key.