Helios Orange a transparentní šifrování (TDE=Transparent Data Encryption) v SQL Serveru 2008+

Trocha teorie na začátek

Transparentní šifrování dat šifruje každou stránku vaší celé databáze a automaticky dle potřeby dešifruje každou stránku během přístupu. Tato funkce vám umožňuje zabezpečit celou vaši databázi, aniž byste se starali o podrobnosti zašifrování na úrovni sloupců.

Výhoda této funkce je v tom, že vám umožní zabezpečit transparentně vaši databázi bez jakýchkoli změn v koncových aplikacích. Transparentní zašifrování dat nevyžaduje žádné místo navíc a může generovat daleko efektivnější plány dotazů než dotazy na zašifrovaná data, protože transparentní zašifrování dat umožňuje systému SQL Server používat řádné indexy.

Slabou stránkou transparentního zašifrování dat je skutečnost, že vyžaduje další režii, protože systém SQL Server musí dešifrovat každou stránku dat při každém dotazu.

Nevýhodou transparentního šifrování dat je to, že s sebou přináší režii v průběhu každého přístupu k databázi, protože kompletně celé stránky dat je potřeba dešifrovat při každém přístupu k databázi. Transparentní šifrování dat také šifruje systémovou databázi tempdb, což může negativně ovlivnit výkon každé další databáze na stejné instanci SQL Serveru.

Co je potřeba mít

- **Microsoft SQL Server 2008 nebo vyšší**
- a to **Enterprise Edition**
  -nebo-
  Enterprise Evaluation Edition (180-denní zkušební verze; licenčně není určena pro produkční prostředí)
  -nebo-
  Developer Edition (vývojářská verze; licenčně není určena pro produkční prostředí)

Na SQL Server doporučujeme nainstalovat nejvyšší dostupný Service Pack.

Proti čemu je to ochrana?

- proti fyzickému odcizení databázových souborů (typicky přípony .MDF/.NDF/ .LDF)
- proti fyzickému odcizení zálohy databáze (typicky přípona .BAK)

Proti čemu to není ochrana?

- proti odposlouchávání komunikace mezi SQL Serverem a klientskými aplikací (k tomu slouží šifrované připojení)

Jak TDE rozchodit se systémem Helios Orange:

Dejme tomu, že máme nainstalovanou instanci **SQL Serveru 2008 Enterprise Edition** a na něm (nezašifrovanou) databázi **Helios001**. Z SQL konzole (Management Studio) postupně spouštíme tyto příkazy:

```
--vytvoření tzv. master key; TDE se přímo netýká, ale
--je součástí šifrovací infrastruktury SQL Serveru
USE master
CREATE MASTER KEY ENCRYPTION BY PASSWORD='VelmiSilneHeslo'
--pokud master key už existuje:
--Server: Msg 15578, Level 16, State 1, Line 1
--There is already a master key in the database. Please drop it before
performing this statement.

--vytvoření certifikátu pro TDE
CREATE CERTIFICATE ServerovyCertifikat WITH SUBJECT='My Database Encryption
Key Certificate'

--důležité - certifikát zazálohovat; soubor je pak potřeba uschovat na
bezpečném (a požárně odděleném) místě
BACKUP CERTIFICATE ServerovyCertifikat TO FILE = 'C:\CertifikatZaloha'

--vytvoření šifrovacího klíče pro databázi
USE Helios001
CREATE DATABASE ENCRYPTION KEY
WITH ALGORITHM = AES_128
ENCRYPTION BY SERVER CERTIFICATE ServerovyCertifikat
--pokud by certifikát nebyl zazálohován ServerovyCertifikat, objeví se
varování:
--Warning: The certificate used for encrypting the database encryption key
has not been backed up. You should immediately back up the certificate and
the private key associated with the certificate. If the certificate ever
becomes unavailable or if you must restore or attach the database on
another server, you must have backups of both the certificate and the
private key or you will not be able to open the database.

--zapnu TDE pro databázi Helios001 (a skrytě pro systémovou
databázi tempdb)
ALTER DATABASE Helios001 SET ENCRYPTION ON

--zašifrování databází probíhá na pozadí, stav lze sledovat
SELECT name, is_encrypted FROM sys.databases
```

Nyní test, že data není možné "ukrást".

```
--databázi zazálohujeme do .bak souboru
BACKUP DATABASE Helios001 TO DISK=N'C:\Helios001.bak'

--zastavíme instanci SQL Serveru
SHUTDOWN
```

A na jiném SQL serveru:

```
--zkusíme připojení databázových souborů .MDF/.LDF
EXEC sp_attach_db N'Helios001', N'C:\Helios001.mdf',
N'C:\Helios001_log.ldf'
--Server: Msg 33111, Level 16, State 3, Line 1
--Cannot find server certificate with thumbprint
'0x679A9C9E0FF58D87B1DB7284A1A3F8CDCED3254E'.

--pokus o obnovu dříve vytvořené zálohy
RESTORE HEADERONLY   FROM DISK=N'C:\Helios001.bak' WITH NOUNLOAD
--ok, zobrazi, že je v .bak souboru obsazena jedna záloha

RESTORE FILELISTONLY FROM DISK=N'C:\Helios001.bak'
--Server: Msg 33111, Level 16, State 3, Line 1
--Cannot find server certificate with thumbprint
'0x679A9C9E0FF58D87B1DB7284A1A3F8CDCED3254E'.
--Server: Msg 3013, Level 16, State 1, Line 1
--RESTORE FILELIST is terminating abnormally.

RESTORE DATABASE Helios001 FROM DISK=N'C:\Helios001.bak' WITH FILE=1,
MOVE N'Helios001'     TO N'C:\Helios001.mdf',
MOVE N'Helios001_log' TO N'C:\Helios001_log.ldf'
--Server: Msg 33111, Level 16, State 3, Line 1
--Cannot find server certificate with thumbprint
'0x679A9C9E0FF58D87B1DB7284A1A3F8CDCED3254E'.
--Server: Msg 3013, Level 16, State 1, Line 1
--RESTORE DATABASE is terminating abnormally.
```

Vidíme, tedy, že ani zálohu, ani databázové soubory nelze "ukrást" a přečíst si jejich obsah. Pokud soubory otevřu hexaeditorem, je vidět, že obsah je zašifrován a není čitelný.

Další zdroje:

V češtině:

- Databázový svět > Šifrování na MS SQL Serveru 2008
- WebovéAplikace.info > Transparentní šifrování v MS SQL 2008

V angličtině:

- DatabaseJournal.com > Test vlivu TDE na výkon SQL Serveru
- MSDN > Database Encryption in SQL Server 2008 Enterprise Edition
- MSDN > Understanding Transparent Data Encryption (TDE)
- MSSQLTips.com > SQL Server 2008 Transparent Data Encryption getting started
- SQL-Server-Performance.com > Transparent Data Encryption
- DatabaseJournal.com > Transparent Data Encryption (TDE) in SQL Server 2008
- Microsoft Knowledge Base > Jak na zálohování certifikátu TDE