PV226/MSSQL
Microsoft SQL Server 2012

Kapitola 4: Správa zabezpečení

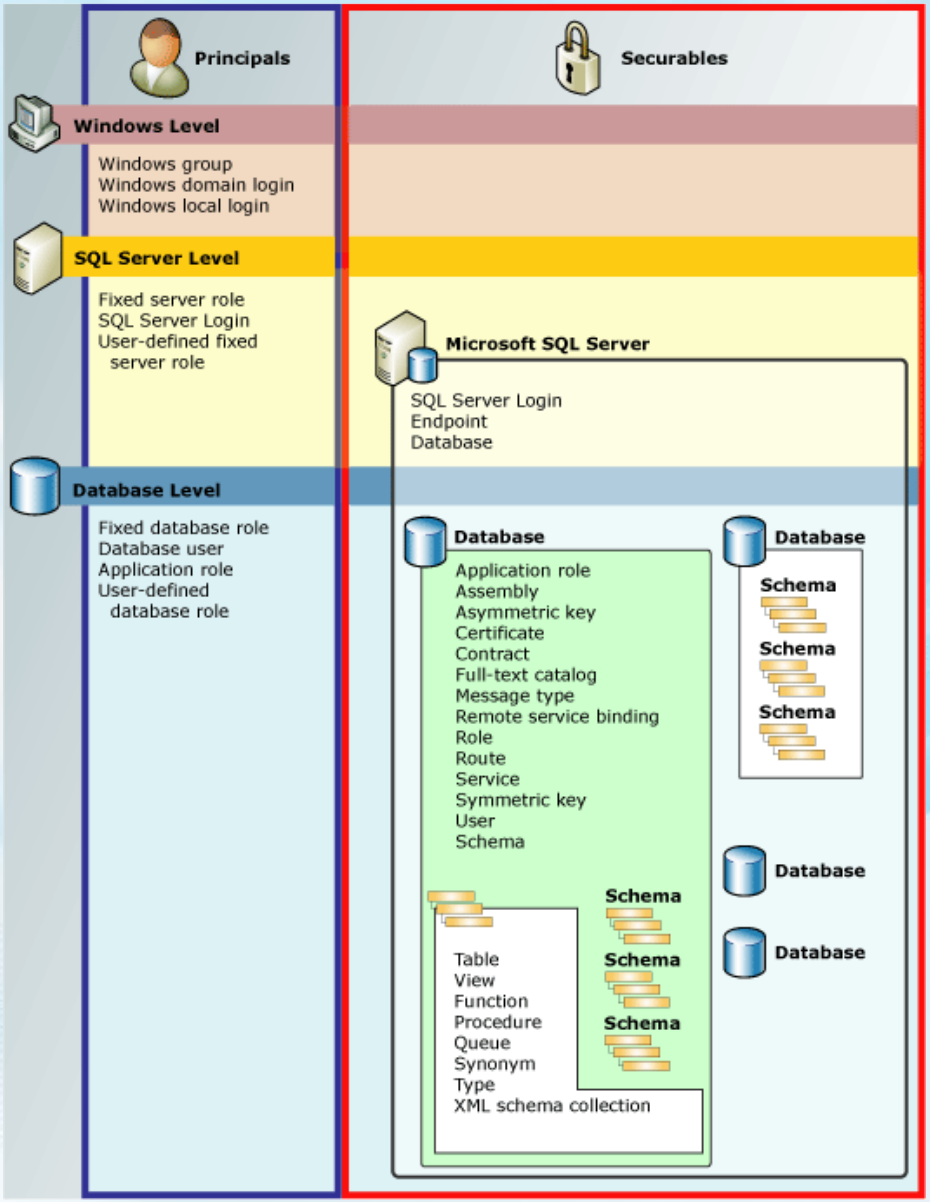
Bc. David Gešvindr
MSP |MCSA | MCTS | MCITP | MCPD

Obsah

1. Části bezpečnostního frameworku
2. Zabezpečení serveru
3. Zabezpečení databáze
4. Správa klíčů a certifikátů

1. Části bezpečnostního frameworku

Hierarchie oprávnění



Uživatel / služba

↓

Ověření **loginu**

↓

V kontextu **DB** je **login** mapován na objekt **user**

Autentizační metody

- Windows Autentication
  - Nezasílá se jméno a heslo při ověření
  - Proces/služba přistupující k SQL je ověřen automaticky operačním systémem
  - **Doporučený postup**
- Mixed SQL and Windows Autentication
  - SQL ověřování kvůli starším aplikacím a scénářům, kde nelze využít Windows ověřování
  - Nevýhodou je vznik většího množství účtů

Execution Context

- Je určen loginem připojeným k dané session
- Autentizační token obsahuje informace o primární a sekundárních identitách
- Určuje práva přístupu k securables objektům v daném spojení
- **Může se během spojení měnit!**

2. Zabezpečení serveru

Serverové role

Role	Popis
sysadmin	Nejvyšší oprávnění
dbcreator	Vytváření a změny databází
diskadmin	Správa datových souborů
serveradmin	Konfigurace nastavení instance
securityadmin	Správa a audit loginů
processadmin	Správa procesů
bulkadmin	Právo pouštět BULK INSERT
setupadmin	Konfigurace replikace a propojených serverů

Správa loginů

Credentials

- Obsahují Windows Autentication informace pro přístup k zdrojům mimo SQL Server
- SQL Login může být svázán jen s jedním objektem Credential

http://msdn.microsoft.com/en-us/library/ms161950(v=sql.110).aspx

3. Zabezpečení databáze

Database Level Principals

- Uživatelé / skupiny kterým lze na úrovni databáze přiřadit oprávnění

- User
  - Uživatel mapovaný na login
- Database Role
  - Skupina uživatelů se stejnými právy
- Application Role
  - Virtuální uživatel do kterého se lze přepnout

Database Roles

- Důležité:
  - db_owner
  - db_datareader
  - db_datawriter

  - http://msdn.microsoft.com/en-us/library/ms189121(v=sql.110).aspx

- **Přiřazujte minimální potřebná práva**

Application Role

- Postup využití aplikační role:
  1. Uživatel spustí aplikaci
  2. Aplikace se připojí k MS SQL jako uživatel
  3. Aplikace se ověří pomocí sp_setapprole
  4. Uživatelský kontext daného spojení se přepne z práv uživatele na práva aplikační role

Speciální uživatelé

- **DBO**
  - Login SA a členové role sysadmin jsou namapováni na tohoto uživatele v každé databázi
- **Guest**
  - Tento účet umožňuje přistoupit k databázi uživatelům bez účtu v té databázi

Oprávnění na úrovni databáze

- Přidělení oprávnění (*Permission*) k jednotlivým objektům v databázi (*Securables*) pro uživatele (*Principal*)
  - Grant/ With Grant / Deny
- Objekty mají svého vlastníka
- Ownership Chain
  - Jiný přístup k vyhodnocování oprávnění pokud volá objekt jiný objekt

Správa uživatelů

1. Vytvoříme login
2. Vytvoříme uživatele na úrovni databáze
3. Přiřadíme oprávnění uživateli

4. Contained Databases

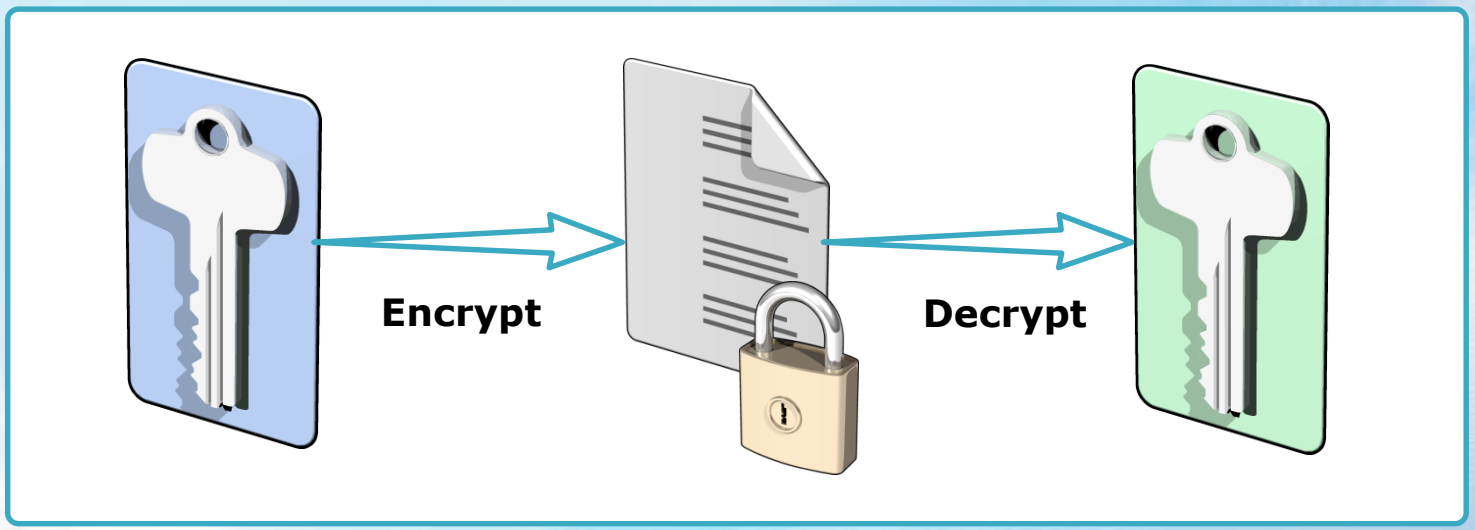
Databáze nezávislá na instanci

- V SQL Serveru 2012 podpora **Partially Contained databází**

- Důležité změny v bezpečnosti:
  - Možnost přihlášení přímo k databázi, obejití instance
  - Správa uživatelů jen v DB
  - Řízení přístupu k DB je zcela v rukou DB_OWNER

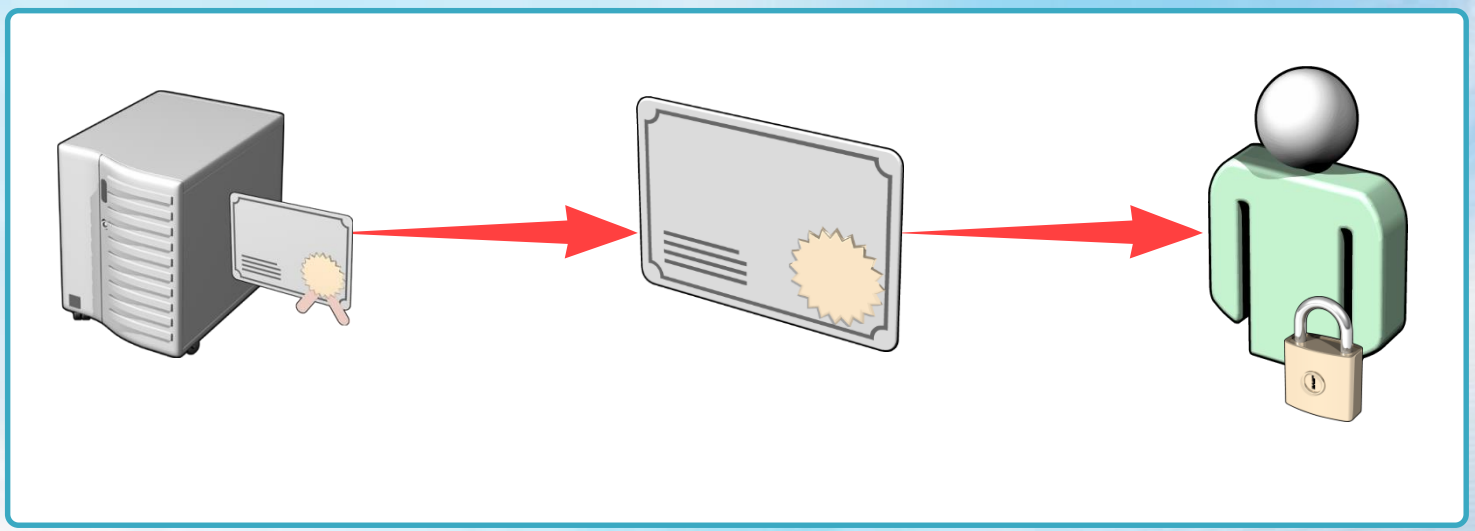
5. Správa klíčů a certifikátů

Klíče



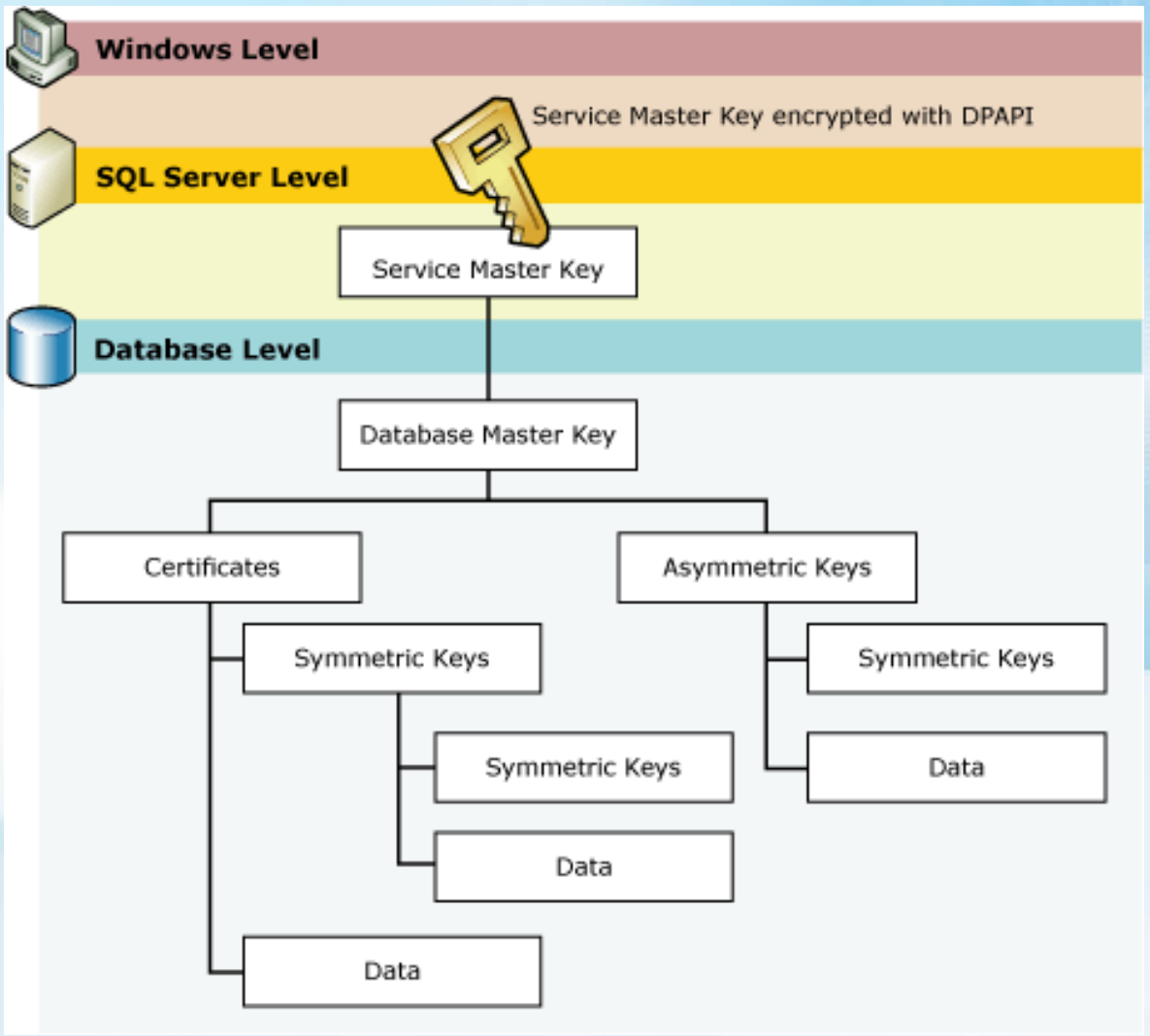
- Symetrické
  - Stejný klíč použit pro šifrování i dešifrování
- Asymetrické
  - Pár klíčů, jeden pro šifrování, druhý pro dešifrování

Certifikáty



- Šifrovací klíče s identifikací vlastníka
  - Veřejný klíč subjektu
  - Identifikační údaje
  - Platnost
  - Identifikace vydavatele
  - Podpis vydavatele

Architektura šifrování



Kdy použít klíče a certifikáty

- Certifikáty
  - Zabezpečení spojení při zrcadlení databáze
  - Podepisování paketů
  - Šifrování spojení
- Asymetrické klíče
  - Zabezpečení uložených dat
  - Zabezpečení symetrických klíčů

Transparent data encryption

- Šifrování dat a transkčního logu v reálném čase

1. Vytvořit „master key“
2. Vytvořit nebo získat certifikát zabezpečený „master key“
3. Vytvořit encryption key a zabezpečit jej certifikátem
4. Povolit šifrování

Transparent Database Encryption Architecture