YAMA

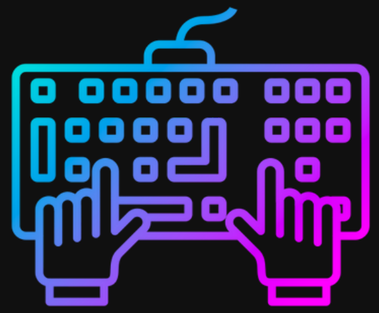
KSE

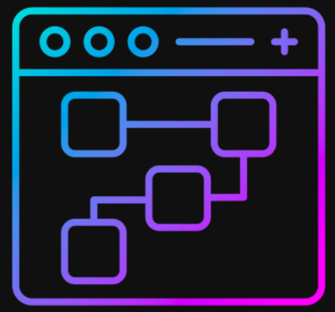
ШИФРУВАННЯ В БАЗАХ ДАНИХ SQL

Мала Ярослава

ЩО ТАКЕ ШИФРУВАННЯ?

Шифрування – це процес перетворення інформації або даних у форму, зрозумілу тільки для авторизованих осіб, що запобігає сторонньому доступу. Це важливий елемент забезпечення конфіденційності та безпеки даних, використовуваний у багатьох сферах, від фінансових транзакцій до особистого спілкування.

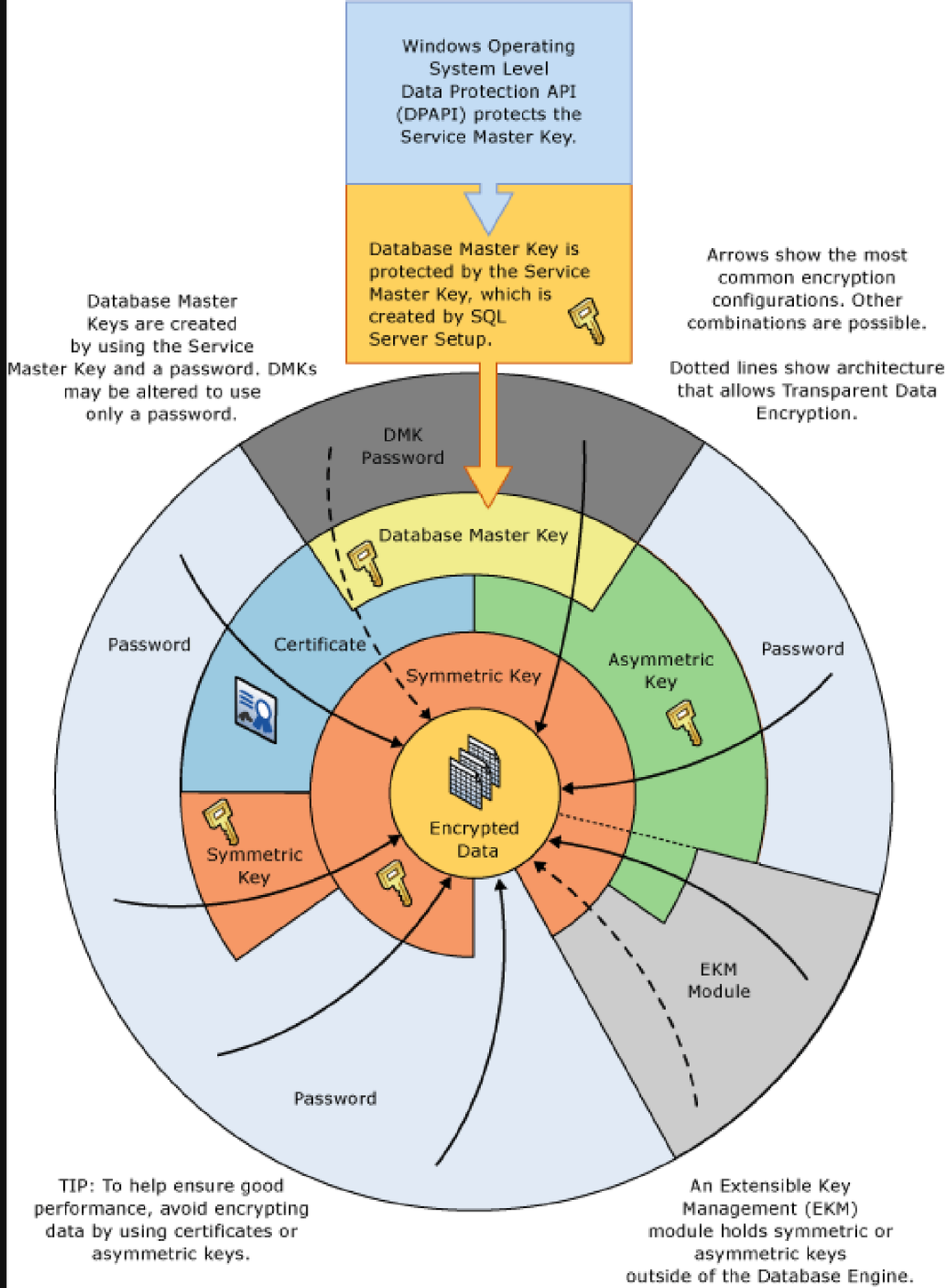




ІЄРАРХІЯ ЗАСОБІВ ШИФРУВАННЯ В SQL SERVER

Yes!

Ієрархія шифрування в SQL Server показує взаємозв'язок між різними ключами та методами, що використовуються для захисту даних. Центральним елементом є Зашифровані дані, які оточені рівнями ключів, що забезпечують їх захист.



МЕХАНІЗМИ ШИФРУВАННЯ

- Функції Transact-SQL;
- Асиметричні ключі;
- Симетричні ключі;
- Сертифікати;
- Прозоре шифрування даних;

ФУНКЦІЇ TRANSACT-SQL ДЛЯ ШИФРУВАННЯ

Функції Transact-SQL для шифрування надають розробникам засоби для впровадження шифрування на рівні програмного забезпечення. Вони дозволяють виконувати шифрування та розшифрування даних безпосередньо в SQL запитах, що дає можливість вбудовувати шифрування в бізнес-логіку додатків.

```
USE AdventureWorks2022;
GO
-- Create a column in which to store the encrypted data.
ALTER TABLE Sales.CreditCard
    ADD CardNumber_EncryptedbyPassphrase VARBINARY(256);
GO
-- First get the passphrase from the user.
DECLARE @PassphraseEnteredByUser NVARCHAR(128);
SET @PassphraseEnteredByUser
    = 'A little learning is a dangerous thing!';

-- Update the record for the user's credit card.
-- In this case, the record is number 3681.
UPDATE Sales.CreditCard
SET CardNumber_EncryptedbyPassphrase = EncryptByPassPhrase(@PassphraseEnteredByUser
    , CardNumber, 1, CONVERT(varbinary, CreditCardID))
WHERE CreditCardID = '3681';
GO
```

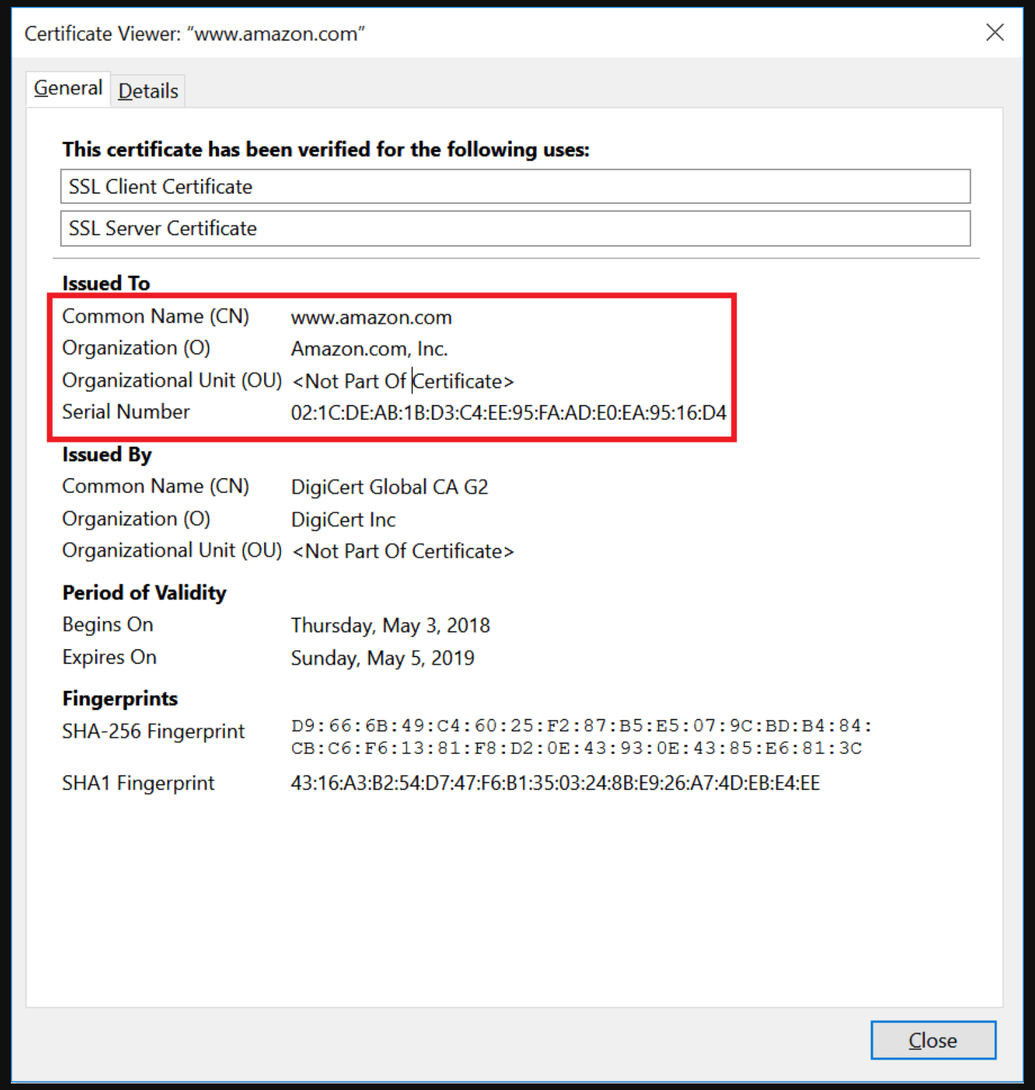
ENCRYPTBYPASSPHRASE (Transact-SQL)

АСИМЕТРИЧНІ КЛЮЧІ

Асиметричні ключі в SQL Server використовуються для шифрування та розшифрування даних за допомогою двох відмінних ключів – одного для шифрування (публічного) та іншого для розшифрування (приватного).

СИМЕТРИЧНІ КЛЮЧІ

Симетричні ключі використовують один і той же ключ для шифрування та розшифрування даних, забезпечуючи баланс між безпекою та продуктивністю.



СЕРТИФІКАТИ

Сертифікат є підписаним цифровим підписом інструкцією, яка пов'язує значення відкритого ключа з ідентифікатором користувача, пристрою або служби, що має відповідний закритий ключ. Сертифікати поставляються та підписуються центром сертифікації (Certification authority, CA).

ПРОЗОРЕ ШИФРУВАННЯ ДАНИХ (TDE)

Прозоре шифрування даних (TDE) у SQL Server надає можливість шифрувати дані "на льоту", забезпечуючи захист усієї бази даних, включаючи файли даних і журналів. TDE працює на рівні файлової системи, шифруючи файли бази даних і автоматично розшифровуючи їх при доступі авторизованим користувачам.

```
USE master;
GO
CREATE MASTER KEY ENCRYPTION BY PASSWORD = '<UseStrongPasswordHere>';
GO
CREATE CERTIFICATE MyServerCert WITH SUBJECT = 'My DEK Certificate';
GO
USE AdventureWorks2022;
GO
CREATE DATABASE ENCRYPTION KEY
WITH ALGORITHM = AES_256
ENCRYPTION BY SERVER CERTIFICATE MyServerCert;
GO
ALTER DATABASE AdventureWorks2022
SET ENCRYPTION ON;
GO
```

ПРИКЛАДИ КЛЮЧІВ ШИФРУВАННЯ

Симетричний ключ (у форматі шістнадцяткового рядка):

```
A3D76C8F7E8D9FAE1B4C2F9D8E7A6B5C
```

Публічний ключ в форматі PEM:

```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAtVKUtcx/n9lFnFyZJ1U+
l+H92q+1F9QZVE4/vyHgA9VhHhjLXe5VE6vhLkARc4LzR7xL7JOZoMmQ211l9Lof
...
-----END PUBLIC KEY-----
```

Серійний номер сертифіката:

```
02:34:56:78:9A:BC:DE:F0:12:34:56:78:9A:BC:DE:F0
```

ДЯКУЮ ЗА УВАГУ!

https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-sql/
https://dou.ua/lenta/articles/searchable-encryption/
https://learn.microsoft.com/ru-ru/sql/relational-databases/security/encryption/sql-server-encryption?view=sql-server-ver16
https://learn.microsoft.com/ru-ru/sql/relational-databases/security/encryption/encryption-hierarchy?view=sql-server-ver16