Encrypting your data through Db2 built-in functions 9.6

Last Updated: 2024-10-29

Db2 provides built-in data encryption and decryption functions that you can use to encrypt sensitive data, such as credit card numbers and medical record numbers. 9.6, 9.7

You can encrypt data at the column or value level. You must install the Integrated Cryptographic Service Facility (ICSF) to use the built-in functions for data encryption.

When you use most data encryption, Db2 requires the correct password to retrieve the data in a decrypted format. If an incorrect password is provided, Db2 does not decrypt the data. If the ENCRYPT_DATAKEY built-in function is used to encrypt data, you must have access to the key label stored with the

|< Db2 for z/OS offers two different sets of built-in functions for encrypting and decrypting data:

- Data encryption using the TDES algorithm: The ENCRYPT_TDES built-in function encrypts data using a password, and the DECRYPT_BIT, DECRYPT_CHAR and DECRYPT_DB built-in functions decrypt data using a password. The DECRYPT_BIT, DECRYPT_CHAR and DECRYPT_DB built-in functions decrypt data that was encrypted with the ENCRYPT_TDES built-in function. The user provides the password when invoking a decryption function
- Data encryption data using the 256-bit AES CBC algorithm: The ENCRYPT_DATAKEY built-in function encrypts data using the 256-bit AES CBC algorithm with either a random initialization vector (IV) or fixed initialization vector (IV) and a key label. The DECRYPT_DATAKEY_INTEGER, DECRYPT_DATAKEY_BIGINT, DECRYPT_DATAKEY_DECIMAL, DECRYPT_DATAKEY_VARCHAR, DECRYPT_DATAKEY_CLOB, DECRYPT_DATAKEY_VARGRAPHIC, DECRYPT_DATAKEY_DBCLOB, AND DECRYPT_DATAKEY_BIT built-in functions decrypt data that was encrypted using the ENCRYPT_DATA KEY built-in function. The decryption process uses the key label stored with the encrypted data to decrypt the data.

>|

Built-in encryption functions work for data that is stored within Db2 subsystem and is retrieved from within that same Db2 subsystem. The encryption functions do not work for data that is passed into and out of a Db2 subsystem. Application Transparent - Transport Layer Security (AT-TLS) is used to encrypt data between Db2 and applications or other data servers.

Attention: When the TDES encryption algorithm is used for encryption, Db2cannot decrypt data without the encryption password. If you forget the encryption password you cannot decrypt the data, and the data might become unusable.

- **Defining columns for data encrypted using the ENCRYPT_TDES built-in function.**
  When data is encrypted using the ENCRYPT_TDES built-in function, it is returned as a binary data

string. Therefore, encrypted data should be stored in columns that are defined as VARCHAR FOR BIT DATA.

- **Defining columns for data encrypted using the ENCRYPT_DATAKEY built-in function.**
  When data is encrypted using the ENCRYPT_DATAKEY built-in function, it is returned as a binary data string. Therefore, encrypted data should be stored in columns that are defined as VARBINARY or BLOB.
- **Defining column-level encryption for the ENCRYPT_TDES built-in function**
  For column-level encryption using the ENCRYPT_TDES built-in function, all encrypted values in a column are encrypted with the same password.
- **Defining the ENCRYPT_TDES function for value-level encryption**
  When you use the ENCRYPT_TDES built-in function for value-level encryption, each value in a given column can be encrypted with a different password. You set the password for each value by using the ENCRYPT keyword with the password.
- **Using predicates for encrypted data**
  When data is encrypted, only = and <> predicates provide accurate results. Predicates such as >, <, and LIKE return inaccurate results for encrypted data.
- **Optimizing performance of data encrypted with the ENCRYPT_TDES function**
  Encryption using the ENCRYPT_TDES function typically degrades the performance of most SQL statements. Decryption requires extra processing, and encrypted data requires more space in Db2.

Parent topic:

→ Protecting data through encryption and RACF

Related concepts

→ Encrypting your data with Secure Socket Layer (SSL) support

→ Protecting data sets through RACF