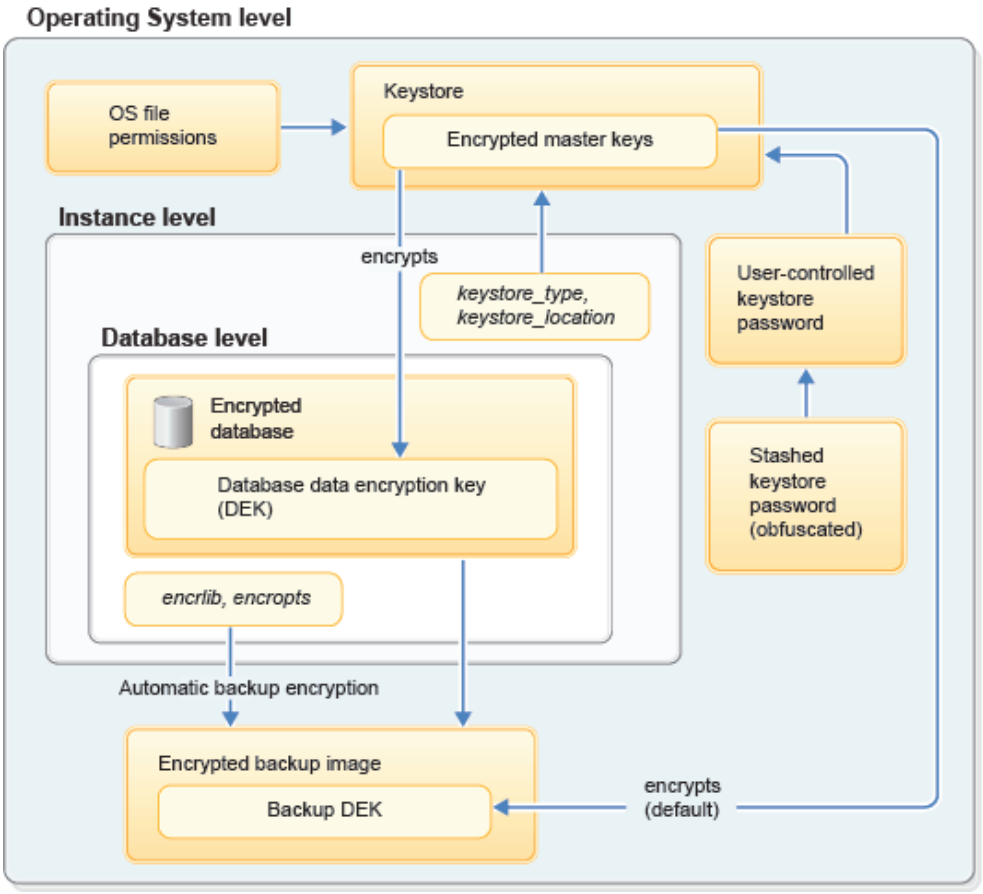

A new master key is automatically added when you create an encrypted database without specifying the MASTER KEY LABEL option on the **CREATE DATABASE** command. The database manager uses this master key by default, but you can optionally add a different master key.
Encrypted master keys are stored in a PKCS#12-compliant *keystore*, which is a storage object for encryption keys that exists at the operating system level. In partitioned database environments or DB2 pureScale® environments, the keystore location must be accessible to all members. There is at most one keystore per DB2 instance. The **keystore_type** and **keystore_location** database manager configuration parameters are used to specify the type and location of the keystore.

The keystore password (in obfuscated form) can be stashed to a file that automatically provides the password when required. The stash file can be read by only the file owner. Not stashing the password enhances security if the instance owner account becomes compromised. However, this additional security must be weighed against any requirements that the DB2 instance can start without human intervention. If the password is not stashed, you cannot access an encrypted database until you provide the keystore password. 26

Database backup images are automatically encrypted if the **encrlib** and **encropts** database configuration parameters are set to a non-null value. The encrypted master key encrypts the backup DEK by default.

DB2 Version 10.5 Fix Pack 5 adds native database encryption to the DB2 database server. This enhancement is easy to implement and provides secure local key management that is based on Public Key Cryptography Standard #12 (PKCS#12). DB2 native encryption enables you to meet compliance requirements in a cost effective manner.

Encrypted backups

Last Updated: 2024-11-14

26

With Db2® native encryption, you can encrypt your database, your database backups, or both. Database backups can be encrypted regardless of whether the database itself is encrypted.

You can encrypt individual backups manually, by specifying the ENCRYPT option on the BACKUP DATABASE command. You can also configure Db2 to automatically encrypt backups by setting the *encrlib* and *encropts* database manager configuration parameters. By default, when an encrypted database is created, these parameters are set to ensure that backups are automatically encrypted. For more information, refer to *Encrypted database backup images*.

26

Important consideration for encrypted backups

When a database backup is encrypted, it is no longer affected by subsequent attempts to reduce its size. Size reduction methods include attempts through compression or data deduplication technologies that are offered on some storage media devices. Encryption removes repetitive patterns from the data that these technologies rely upon. To reduce the size of database backups, compression needs to be applied before encryption. Compression can be done by actively compressing the data in the database itself, or by specifying the `libdb2compr_encr.so` or `libdb2nx842_encr.a` library on the BACKUP DATABASE command.

– **Encrypted database backup images**
  You can create an encrypted backup image of your database, then retrieve it using the **RESTORE DATABASE** or **RECOVER DATABASE** command. **RECOVER DATABASE** runs both the **RESTORE DATABASE** and **ROLLFORWARD DATABASE** command.

Parent topic:

→ Db2 native encryption

Related information

→ **BACKUP DATABASE command**