DATABASE

第5章

暗号化，レプリケーション，高可用性

データベースの安全性確保の手法

David Zhang

データベースは，データの安全性を確保するためにさまざまな手法を用いています．ここではまず，データベースの安全性を確保するために必要な暗号化の手法について解説します．次にデータベースのレプリケーション（複製）と高可用性（HA ： High Availability）の概念について解説を行います．

（編集部）

1 一般的なデータベースにおけるデータ保護の手法

データベースの暗号化の概念を**図1**に示します．

● ログイン名とパスワードで保護されたデータベース

通常，Windows あるいは Linux では，マシンごとにログイン ID とパスワードをもっています．これと同じ手法を利用してデータベースを扱うユーザを登録することにより，データベースのフォルダやデータ・ファイルを保護できます．しかし，この方法はデータベース保護の手法としては弱いものです．

● ユーザ権限の管理

データベースでは，保存されているデータに対するユーザのアクセス権を管理します．組み込みデータベースは，ユーザあるいはアプリケーションに対して，データベースの中のテーブルやカラムのアクセス権を与えたり，はく奪したりできます．

組み込みデータベースには，参照権限，更新権限，削除権限，挿入権限などがあります．それらの権限をうまく使うことで，データベースにアクセスできるユーザを制限したり，いろいろな条件でユーザを選別することができます．

ここでは Empress 社の組み込みデータベースを例に，アクセス権管理の方法を示します．テーブル t への参照である SELECT 権を DAVID に与えるコマンドは，以下のとおりです．

```
grant select on t to DAVID as dba
```

また，DAVID の権限を取り除くコマンドは，以下のとおりです．

```
revoke select on t from DAVID as dba
```

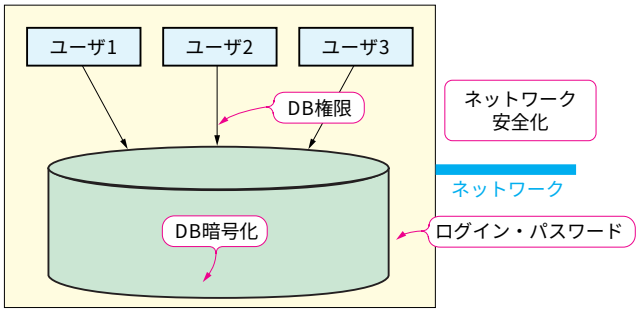

図1　データベースの暗号化の概念

● ネットワークの安全性

ネットワークに接続しているシステムの中のデータベースは，ネットワーク上でデータの転送を行います．そのため，ネットワーク上の通信の安全性を確保する必要があります．ただし，この安全性はネットワークの安全性であり，組み込みデータベースの担当範囲外です．別途，ネットワークの部分で暗号化などの安全性確保の手法が必要になります．

2 データベースの暗号化技術

● セキュリティの重要性

データ・セキュリティを強化することは，官公庁や企業にとって最優先課題となってきています．企業の従業員が機密情報を持ち出した，といった話題を聞かない日はありません．最近では，多くの国で法制度が整備され，医療記録の保護や，プライバシの保護，個人の貯蓄情報の保護など，さまざまなセキュリティの基準が策定されつつあります．

このような背景から，コンピュータが扱う情報を保護することは必須となりつつあります．この中には，データ・セキュリティのしくみによってデータを保護するというだけでなく，アプリケーションやネットワークに対する攻撃を阻止するという意味も含まれます．

また，暗号化などの新たな技術の進歩によって，データ保護を目的とする，強固なセキュリティ手法が提案されています．

● 安心できるデータ保護

ここで議論するセキュリティは，データベースに保存されているデータを，正しく安心できるレベルで保護する方法です．ほとんどのデータベースは，ハードディスクのような（不揮発

KEYWORD ―データベース，セキュリティ，暗号化処理，レプリケーション，高可用性

表1　暗号に使われる用語の解説

用　語		解　説
暗号アルゴリズム（Cipher ： Cryptographic Algorithm）		暗号化，復号化に使用される数学的な関数
暗号化（Encryption）		その実体を隠すために，暗号という方法でデータを隠ぺいする処理
暗号鍵（Cipher Key）		暗号アルゴリズムと連結して使用し，暗号を決定するパラメータ
	Cryptographic Key	平文データを暗号文データに変換（暗号化）する鍵
	Encryption Key	暗号文データを平文データに変換（復号化）する鍵
暗号文（Ciphertext）		すでに暗号化された一連のデータ
パディング（Padding）		通常，平文が短い場合に付け加えられる文字列．たとえば，データ長が4バイトで暗号化ブロックのデータ長が16バイトの場合，12バイトのパディング・データを付け加える必要がある．パディングされる文字は‘0’か，そのかわりに‘1’，あるいは，ほかの任意のデータ・パターン
平文（Plaintext, Cleartext）		暗号化されていない一連のデータ
復号化（Decryption）		暗号文を変換して平文に戻す処理

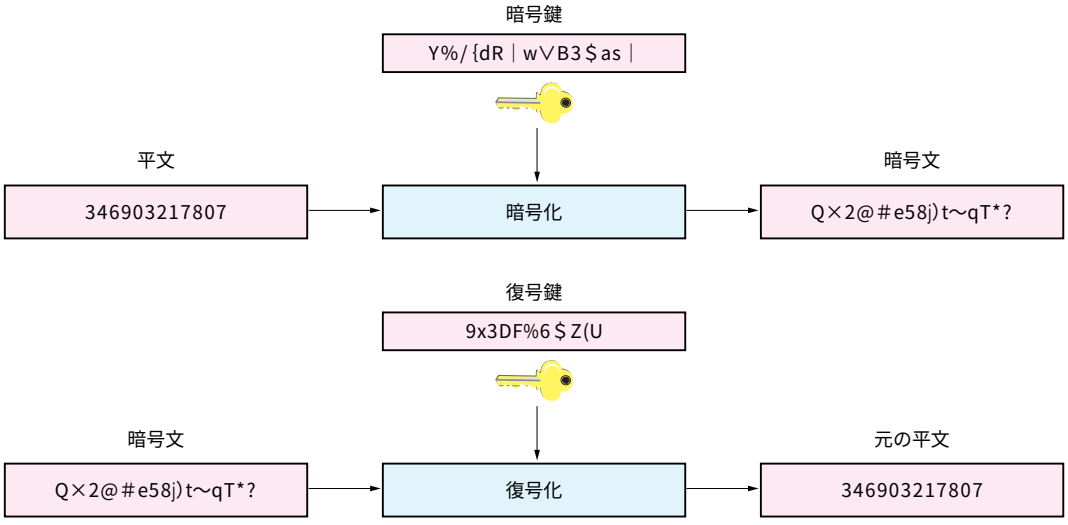

図2　暗号鍵と復号鍵

性の）記憶装置に保存され，配置されます．

メモリ上で動作するデータベースの場合，頻繁にデータをバックアップするために，同じような記憶装置が必要です．したがって，データは，最後には記憶装置に平文で（オリジナルのまま）保存されることになります．つまり，組み込みデータベースを実装している多数の装置は，保護されるべき機密事項に類するデータを保持していることが多いといえます．

このようなデータを保持するためには暗号化技術の採用が必須です．しかし，データベースに暗号化技術を実装する際には，多くの重要な問題を考慮しなければなりません．

たとえば，以下のような問題があります．

- 暗号化処理は，データベース・エンジンの内部で実行されるべきか，データを生成するアプリケーションで実行されるべきか．それとも，ハードウェアで実行されるべきか
- 暗号鍵は，データベースの内部に保存されるべきか，もっと安全などこかほかの場所に保存されるべきか
- データ暗号化の粒度はデータベース単位か，テーブル単位か，あるいはカラム単位か
- 性能とメモリ容量を考えたとき，ほかにどのようなアプローチがありえるか

などです．

本稿ではこのような課題について検討します．そしてEmpress社の組み込みデータベースに搭載された暗号化技術を例に，その実装方法を解説します．最後にユーザにとっての利点についてまとめます．

本稿で使用する基本的な用語を**表1**に示します．

また，鍵ベースのアルゴリズムには「公開鍵方式」と「対称鍵方式」があります．**図2**に公開鍵方式を用いてデータを暗号化する処理と，復号化する処理について示します．この場合，暗号鍵と復号鍵は別のものになります．

対称鍵方式の場合，暗号鍵と復号鍵は同じです．今回，例として取り上げるEmpress社の組み込みデータベースは対称鍵