SQL Anywhere のテーブルの暗号化

この技術ドキュメントでは、暗号化の基礎と、SQL Anywhere 10 データベースでテーブルの暗号化を設定する方法について説明します。

はじめに

暗号化は、情報の閲覧を許可されていないユーザから情報を保護するために使用するセキュリティ・ツールです。SQL Anywhere のバージョン 8.0.0 では、強力な暗号化が採用されました。SQL Anywhere のバージョン 10.0.0 では、データベース全体、または選択したテーブルのみを暗号化できます。これによりさらに柔軟性が向上し、データベースの一部のみを暗号化すればよい場合に性能を向上できます。

このドキュメントは、Microsoft Windows XP 上で動作する SQL Anywhere 10.0.0.2788 の場合について説明しています。ただし、Linux や Solaris などの他のシステム上の SQL Anywhere 10 にも適用できます。

暗号化の概要

暗号化とは、情報の閲覧を許可されていないユーザに対し、その情報の解読を困難にするプロセスです。暗号化されたデータベース・ファイルに物理的にアクセスできても、ファイルを調べてデータベースに含まれるデータを表示することはできません。

暗号化には、簡易暗号化と強力な暗号化の 2 種類があります。簡易暗号化では、データを難読化しますが、暗号化キーは使用しません。許可されていないユーザがディスク・ユーティリティを使用してデータを表示することを、防止する簡単な方法です。強力な暗号化では、暗号化アルゴリズムを使用してデータを暗号化し、キーを持つユーザのみにアクセスを許可します。

暗号化方式とは、情報の暗号化と復号化に使用するアルゴリズムです。暗号化方式を使用して、読みやすいプレーン・テキストを人間には判読不能な暗号文に変換します。ブロック暗号は、プレーン・テキストのブロックを暗号文のブロックに変換します。ブロックとは、固定バイト数のテキストの集合です。

SQL Anywhere の暗号化の概要

SQL Anywhere 10 は、強力な暗号化用に AES 暗号化方式を実装しています。このアルゴリズムは Rijndael と呼ばれ、アメリカ政府により次世代標準暗号化方式として選択されています。

SQL Anywhere の一部のプラットフォーム、特に Windows x86 や Windows CE では、AES-FIPS 暗号化方式も使用できます。AES-FIPS は、基本的には AES と同一です。唯一の違いは、AES-FIPS では Certicom による実装を使用している点です。この実装は、FIPS 規格に従い、アメリカ政府により承認されています。FIPS は、Federal Information Processing Standard (連

邦情報処理規格）の略語です。SQL Anywhere 10 では、データベースのページ・サイズと同じサイズのブロックが使用されます。SQL Anywhere 10 での AES および AES-FIPS の実装では、128 ビット・ブロックを使用しています。

暗号化キーの指定

強力な暗号化を使用する場合、使用する暗号化キーを指定する必要があります。強力な暗号化で使用するキーを作成する場合、簡単に推測できないキーを作成することが重要です。キーが長いほど、推測が困難になります。暗号化キーは、大文字、小文字、数字、特殊文字で構成できます。暗号化キーを覚えておくことが非常に重要です。キーを失くすと、データベースは永久にアクセスできなくなります。

暗号化キーの変更

Interactive SQL で CREATE ENCRYPTED FILE 文を使用して、暗号化キーを変更できます。この SQL 文は、新しい暗号化キーを設定した新しいデータベースを作成することで、暗号化キーを変更し、既存のデータベースを新しいデータベースにロードします。

次の SQL 構文は、oldfile.db という既存のデータベースから newfile.db という新しいデータベースを作成します。新しい暗号化キーは new_key で、古い暗号化キーは old_key です。また、使用する暗号化アルゴリズムは AES です。

```
CREATE ENCRYPTED FILE ‘c:\newfile.db’
FROM ‘c:\oldfile.db’
KEY ‘new_key’
OLD KEY ‘old_key’
ALGORITHM ‘AES’;
```

この文を、トランザクション・ログ・ファイルと、すべての dbspace またはミラー・ログ・ファイルに対しても実行する必要があります。

暗号化を有効にする

データベース内のテーブルで暗号化を行うには、データベースの作成時に暗号化を有効にする必要があります。暗号化を有効にする際、簡易暗号化と強力な暗号化のどちらを使用するかを決定する必要があります。

テーブルの暗号化を有効にする

テーブルに簡易暗号化を指定してデータベースを作成するには、次の手順に従います。 （コマンドプロンプト）

1. コマンド・プロンプトで次のコマンドを実行します。

dbinit -e -et *demo.db*
このコマンドは、dbinit **ユーティリティを使用して新しいデータベースを作成します。このコマンドは、簡易暗号化** (-e)**とテーブルの暗号化** (-et)**を有効にして、新しいデータベースの名前を** demo.db **に設定します。**

テーブルに簡易暗号化を指定してデータベースを作成するには、次の手順に従います。(SQL)

1. Interactive SQL **を起動します。**
2. **以下の** SQL **文を実行します。**

```
CREATE DATABASE ‘ c:\demo.db ’
ENCRYPTED TABLE ALGORITHM ‘ simple ’;
```

この CREATE DATABASE **文は、テーブルに簡易暗号化を指定して新しいデータベースを作成します。**

強力な暗号化を有効にする

テーブルに強力な暗号化を指定してデータベースを作成するには、次の手順に従います。(コマンドプロンプト)

コマンド・プロンプトで次のコマンドを実行します。

```
dbinit -ea AES -et -ek myencryptionkey c:\demo.db
```

このコマンドは、dbinit **ユーティリティを使用して新しいデータベースを作成します。このコマンドは、強力な** AES **暗号化**(-ea AES)**とテーブルの暗号化** (-et)**を有効にして、暗号化キー** (-ek myencryptionkey)**を指定し、新しいデータベースの名前を** demo.db **に設定します。**

テーブルに強力な暗号化を指定してデータベースを作成するには、次の手順に従います。(SQL)

1. Interactive SQL **を起動します。**
2. **以下の** SQL **文を実行します。**

```
CREATE DATABASE ‘ c:\demo.db ’
ENCRYPTED TABLE KEY ‘ myencryptionkey ’
ALGORITHM ‘ AES ’;
```

この CREATE DATABASE **文は、テーブルに強力な** AES **暗号化を指定して新しいデー**

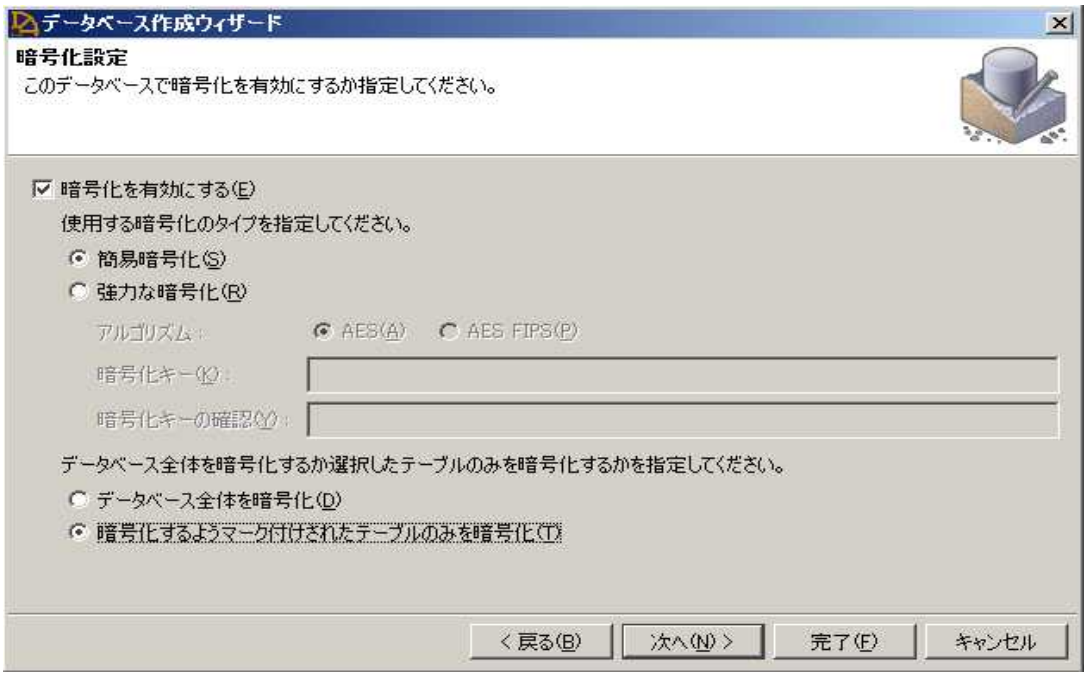
タベースを作成し、暗号化キー（myencryptionkey）を指定します。

Sybase Central からテーブルの暗号化を有効にする

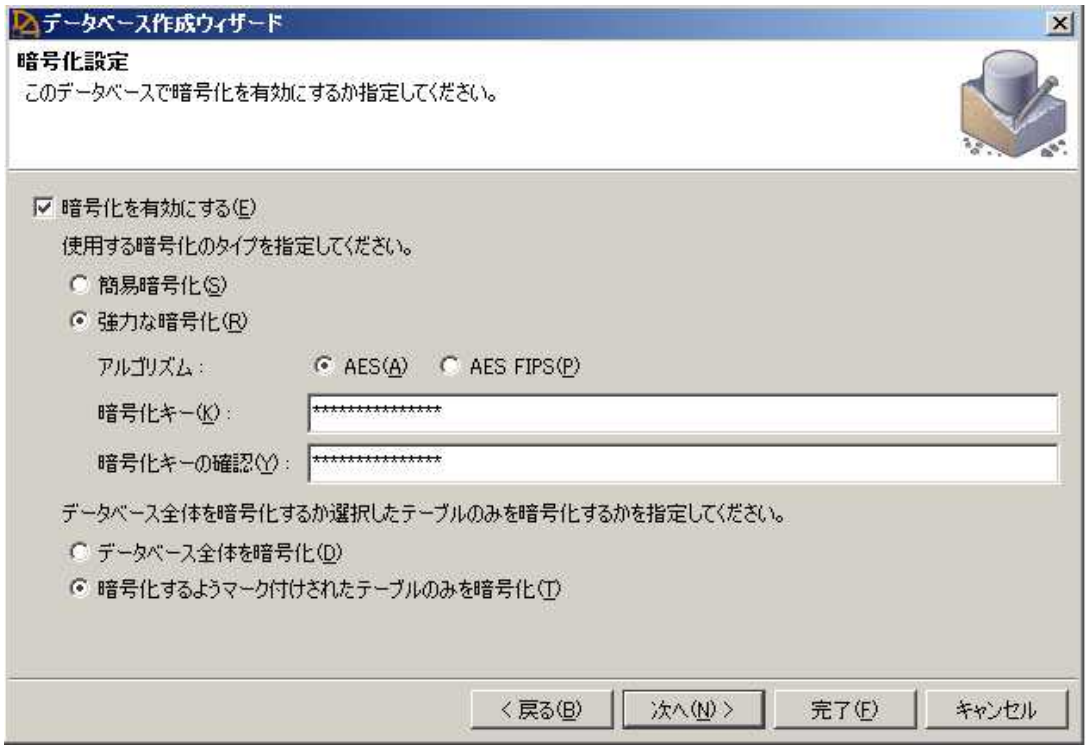
Sybase Central から強力な暗号化されたデータベースを作成するには、次の手順に従います。

1. Sybase Central では、[ツール] メニューから [データベースの作成] を選択します。ウィザードの指示に従います。
2. ウィザードの [暗号化設定] ページで、データベースの暗号化を設定します。

a.テーブルの簡易暗号化を使用するには、[暗号化を有効にする] チェックボックスをオンにして [簡易暗号化] を選択し、暗号化キーを指定して [暗号化するようにマーク付けされたテーブルのみを暗号化] を選択します。



b. テーブルの強力な暗号化を使用するには、[暗号化を有効にする] チェックボックスをオンにして [強力な暗号化] を選択し、暗号化キーを指定して [暗号化するようにマーク付けされたテーブルのみを暗号化] を選択します。



テーブルの暗号化

テーブルは、作成時に暗号化します。
テーブルを暗号化するには、次の手順に従います。(Interractive SQL)

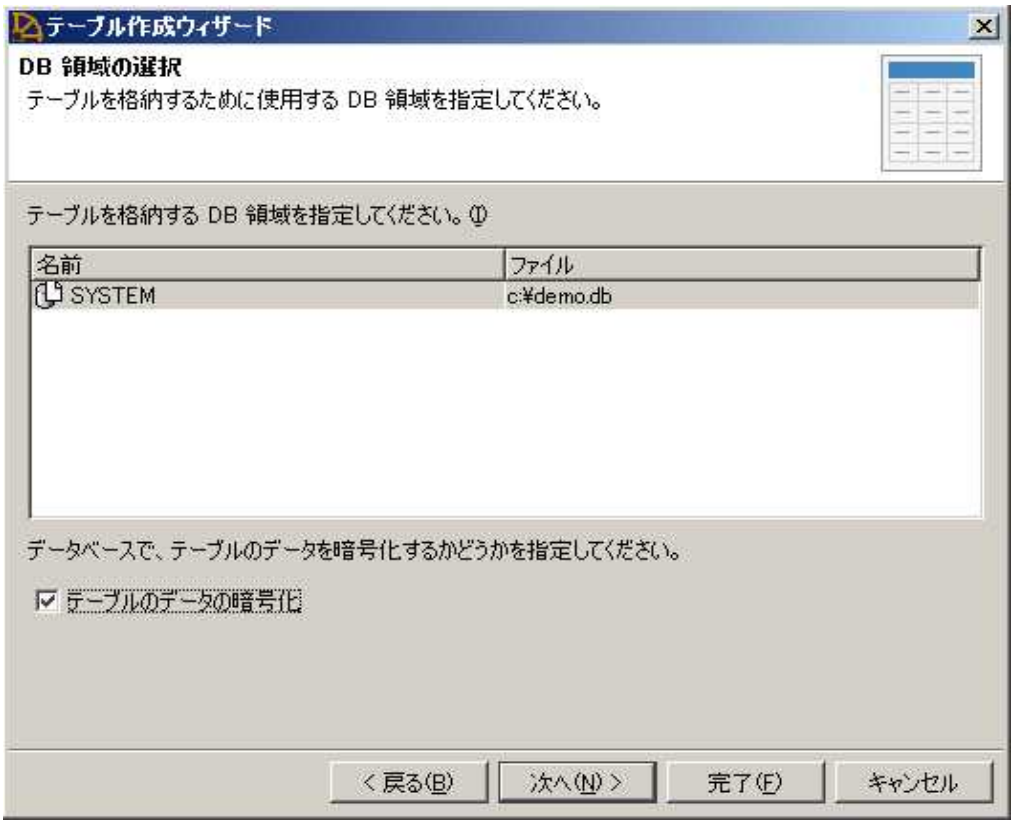
1. Interactive SQL を起動し、データベースに接続します。
2. 次のコマンドを実行します。

```
CREATE TABLE Employees (
UserName CHAR( 50 ),
UserID INTEGER )
ENCRYPTED;
```

テーブルを暗号化するには、次の手順に従います。(Sybase Central)

1. Sybase Central を起動し、データベースに接続します。
2. Tables フォルダを選択します。
3. [ファイル] ー [新規] ー [テーブル] を選択します。
   [テーブル作成] ウィザードが表示されます。
   [テーブルのデータの暗号化] オプションが選択されていることを確認し、ウィザードの指示に

従います。



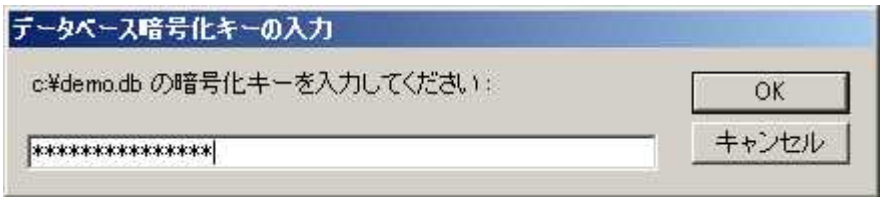
暗号化されたデータベースのアクセス

簡易暗号化が使用されている場合、またはテーブルの簡易暗号化が使用されているデータベースを開始する場合は、キーは必要ありません。簡易暗号化ではキーを使用しないため、データベースまたはテーブルのデータには接続パラメータだけでアクセスできます。データベース･ファイルそのものは解釈不能になっています。

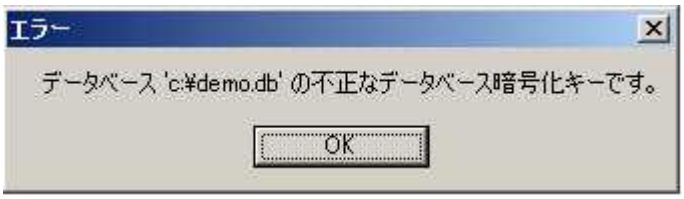
強力な暗号化が使用されているデータベース、またはテーブルに強力な暗号化が使用されているデータベースを開始する場合は、暗号化キーを指定する必要があります。指定するには、データベース･サーバ開始時に `-ep` **または** `-ek` **オプションを使用します。**

`-ep` **オプションを使用した場合は、暗号化キー入力用ウィンドウが表示されます。このキーは、プレーン･テキストでは表示されません。**

```
dbeng10 -ep c:\demo.db
```



誤ったキーを入力すると、エラー・メッセージが表示され、データベース・サーバが終了します。



-ek オプションを使用して暗号化キーを指定することもできます。-ek オプションは、データベース・サーバ・コマンドでデータベース・ファイル名の後に使用します。暗号化キーは、プレーン・テキストで入力します。

```
dbeng10 c:\demo.db  -ek myencryptionkey
```

別の方法として、[Sybase Central 接続] ダイアログを使用してデータベースに接続することもできます。接続パラメータを入力する際に、[データベース] タブでデータベースの暗号化キーを指定できます。