Transparent Data Encryption for PostgreSQL Enterprise Edition
セットアップカード

（Windows 版）

ＯＳＳＤＢＴＤＥ０６－０３

ご注意

1. 本書の内容の一部または全部を無断転載することは、禁止されています。
2. 本書の内容に関しては将来予告なしに変更することがあります。
3. 本書の内容について万全を期して作成いたしましたが、万一ご不審な点や誤り、記載漏れなど、お気付きのことがありましたらご連絡ください。

輸出する際の注意事項

本製品（ソフトウェア）は、外国為替管理令に定める提供を規制される技術に該当致しますので、日本国外へ持ち出す際には日本国政府の役務取引許可申請等必要な手続きをお取りください。

許可手続き等にあたり特別な資料等が必要な場合には、お買い上げの販売店またはお近くの当社営業拠点にご相談ください。

はしがき

このたびは、Transparent Data Encryption for PostgreSQL Enterprise Edition をお買い上げいただき、誠にありがとうございます。

本書は、Transparent Data Encryption for PostgreSQL を使用した透過的暗号化機能の導入を行うエンジニアを対象読者とし、Transparent Data Encryption for PostgreSQL のインストール、アップグレード、アンインストールの手順について説明します。なお、透過的暗号化機能をご使用の際は、さらに『透過的暗号化機能利用の手引』をご確認ください。

重要

本手順書に記載された方法以外でインストールおよびアンインストールを行った場合は、動作の保証はいたしません。

備考

1. 本書に説明しているすべての機能はプログラムプロダクトであり、次のプロダクト型番に対応しています。

プロダクト型番	プロダクト名	対応モデル
UL1298-H001	Transparent Data Encryption for PostgreSQL Enterprise Edition V1.2 Windows 版 (1CPU)(1 年間)	64 ビット
UL1298-H002	Transparent Data Encryption for PostgreSQL Enterprise Edition V1.2 Windows 版 1CPU 追加(1 年間)	64 ビット
UL1298-H003	Transparent Data Encryption for PostgreSQL Enterprise Edition V1.2 Windows 版 Cluster Option(1 年間)	64 ビット
UL1298-H011	Transparent Data Encryption for PostgreSQL Enterprise Edition V1.2 Windows 版 (1CPU)(3 年間)	64 ビット
UL1298-H012	Transparent Data Encryption for PostgreSQL Enterprise Edition V1.2 Windows 版 1CPU 追加(3 年間)	64 ビット
UL1298-H013	Transparent Data Encryption for PostgreSQL Enterprise Edition V1.2 Windows 版 Cluster Option(3 年間)	64 ビット

2. Linux は、Linus Torvalds 氏の米国およびその他の国における登録商標または商標です。
3. Microsoft、Windows、Windows Server、Windows PowerShell は、米国 Microsoft Corporation の米国およびその他の国における登録商標です。
4. Amazon Web Services およびすべての AWS 関連の商標、ならびにその他の AWS のグラフィック、ロゴ、ページヘッダーボタンアイコン、スクリプト、サービス名は、米国および／またはその他の国における、AWS の商標、登録商標またはトレードドレスです。
5. その他、記載されている会社名および製品名は、一般的にそれぞれ各社の商標または登録商標です。

本書の表記規則

本書では、注意すべき事項、重要な事項および関連情報を以下のように表記します。

注

この表記は、重要であるがデータ損失やシステムおよび機器の損傷には関連しない情報を表します。

重要

この表記は、データ損失やシステムおよび機器の損傷を回避するために必要な情報を表します。

ヒント

この表記は、お客様に役立つ可能性のある情報を表します。

実行例およびファイルの設定例は以下のように表記します

```
コマンドラインの実行例を示します
```

```
ファイルの設定例を示します
```

また、本書では以下の表記法を使用します。

表記	使用方法	例
コマンドライン中の [ ] 角かっこ	かっこ内の値の指定が省略可能であることを示します	cipher_setup.sh [-s {1|2} [path] | -h］
コマンドライン中の { } 波かっこ	かっこ内の値のいずれかを指定する必要があることを示します	cipher_setup.sh [-s {1|2} [path] | -h] 上記例の場合角かっこ内に波かっこがあるため、"-s" オプションを指定した場合、 "1" または "2" を指定する必要があります
#	OS の管理者ユーザで発行するコマンドを示すプロンプトです	# `./cipher_setup.sh`
$	OS の一般ユーザ（postgres など）で発行するコマンドを示すプロンプトです	$ `psql`
=#	PostgreSQL のスーパーユーザーで SQL を発行する場合は、「=#」のように表記しますが、明示的に接続しているデータベース名を示す場合は、「postgres=#」や「testdb=#」のように先頭にデータベース名を含みます	=# SELECT count(*) FROM public.cipher_key_table;
=>	PostgreSQL の一般ユーザーで SQL を発行する場合は、「=>」のように表記しますが、明示的に接続しているデータベース名を示す場合は、「postgres=>」や「testdb=>」のように先頭にデータベース名を含みます	=> SELECT c1 FROM t1;
CMD>	Windows のコマンドプロンプトで発行するコマンドを示します	CMD>`ipconfig`
モノスペースフォント斜体	ユーザーが有効な値に置き換えて入力する項目	tde_for_pg<*PostgreSQL メジャーバージョン*>-<*Transparent Data Encryption for PostgreSQL バージョン*>-.<*Red Hat Enterprise Linux バージョン*>.x86_64.rpm

最新情報の入手先

最新の製品情報については、以下の Web サイトを参照してください。

https://jpn.nec.com/tdeforpg/

目次

第 1 章
はじめに

本章では、Transparent Data Encryption for PostgreSQL の紹介と Edition ごとの提供機能やサービスについて説明します。

1.1　Transparent Data Encryption for PostgreSQL とは

Transparent Data Encryption for PostgreSQL を使用することで、表に格納する機密データを暗号化できます。また、暗号化されたデータを処理するアプリケーションは、ほとんどあるいはまったく変更せずに透過的にデータを暗号化、復号することができます。さらに、暗号鍵の管理を簡単に行う機能も提供するサブスクリプション製品です。

1.2　Edition ごとの利用可能な機能と提供されるサービス

Transparent Data Encryption for PostgreSQL には、商用版の Enterprise Edition と OSS として公開している Free Edition があります。 各 Edition で利用可能な機能と提供されるサービスを示します。

表 1-1　Edition による機能/サービスの違い

機能/サービス		Enterprise Edition for Linux	Enterprise Edition for Windows	Free Edition
Transparent Data Encryption 機能				
列単位の暗号化機能	テキスト	○	○	○
	バイト列（画像など）	○	○	○
	NUMERIC	○	○	×
	整数型（smallint,integer,bigint）	○	○	×
	日付・時刻	○	○	×
鍵の更新、バージョン管理機能		○	○	△*1
AWS Key Management Service を利用した鍵管理		○	×	×
簡易 TDE モード		○	○	×
サポートサービス				
Transparent Data Encryption for PostgreSQL の PP サポートサービス		○	○	×
PostgreSQL 本体の保守サポートサービス		○	○	×

*1　暗号鍵のバージョン管理機能なし。一括更新のみ可

第 2 章
インストールの概要

本章では、Transparent Data Encryption for PostgreSQL のインストール、アップグレード、アンインストールの概要について説明します。

2.1　インストールの種類

本書で説明する Transparent Data Encryption for PostgreSQL のインストールの種類は以下の 2 つがあります。

- 新規インストール

  Transparent Data Encryption for PostgreSQL を初めてインストールする場合に行います。

- ストリーミングレプリケーション構成への新規セットアップ

  PostgreSQL の標準機能である、ストリーミングレプリケーションを利用した環境に Transparent Data Encryption for PostgreSQL を初めてインストールする場合に行います。

2.2　アンインストールの種類

本書で説明する Transparent Data Encryption for PostgreSQL のアンインストールには以下の 2 つがあります。

- アンインストール

  Transparent Data Encryption for PostgreSQL をアンインストールする場合に行います。

- ストリーミングレプリケーション構成からのアンインストール

  PostgreSQL の標準機能である、ストリーミングレプリケーションを利用した環境から Transparent Data Encryption for PostgreSQL をアンインストールする場合に行います。

第 3 章
動作環境の確認とインストール前の準備

本章は、Transparent Data Encryption for PostgreSQL を使用するために必要な動作環境とインストール前に確認しておくべきことについて説明します。

3.1　PostgreSQL のインストール

Transparent Data Encryption for PostgreSQL を利用するためには、事前に PostgreSQL をインストールしておく必要があります。「3.3.1.2　ソフトウェア要件（5 ページ）」の条件を満たす PostgreSQL バージョンをインストールしてください。

また、インストール後システム変数 PATH に PostgreSQL の bin を設定します。次の例ではコマンドプロンプトより、システム変数を確認しています。また、PostgreSQL は `C:\Program Files\PostgreSQL\9.6` にインストールされていることとします。

```
CMD>echo %PATH%
C:\Program Files\PostgreSQL\9.6\bin;...
```

3.2　透過的暗号化機能をセットアップするために必要な情報

透過的暗号化機能をセットアップするために必要な PostgreSQL の接続情報を確認します。

表 3-1　透過的暗号化機能をセットアップするために必要な PostgreSQL の接続情報

ポート番号	透過的暗号化機能をセットアップするデータベースが定義された PostgreSQL のサービス待ち受けポート番号です。
データベース名	透過的暗号化機能をセットアップするデータベースの名前です。
スーパユーザ名	透過的暗号化機能をセットアップするデータベースに接続するためのスーパユーザです。
スーパユーザのパスワード	透過的暗号化機能をセットアップするデータベースに接続するためのスーパユーザのパスワードです。
セキュリティ管理ユーザ名	透過的暗号化機能の暗号鍵を管理するための専用のユーザです。
セキュリティ管理ユーザのパスワード	透過的暗号化機能の暗号鍵を管理するための専用のユーザのパスワードです。

重要

禁則文字

本ツールで構築する透過的暗号化環境の中で使用する次のオブジェクトでは、「機種依存文字」「Unicode の重複文字」「改行文字」「空文字」の使用を禁止しています。また、個々のオブジェクトで使用を禁止している文字・文字列は次の通りです。

- ホスト名

  {「!」,「'」}を同時使用,「マルチバイト文字」の使用を禁止しています

- データベース名

  {「!」,「'」}を同時使用,「'」,「"」,「/」,「¥」,「=」,「:」,「?」「マルチバイト文字」の使用を禁止しています。

  複数のデータベースインスタンス（データベースクラスタ）を同時に使用する場合、データベース名が重複 しないようご注意ください。

- ユーザ名

  {「!」,「'」}を同時使用,「'」,「"」,「マルチバイト文字」の使用を禁止しています。

- パスワード

  {「!」,「'」}を同時使用,「マルチバイト文字」の使用を禁止しています。

表 3-2　接続情報禁則文字一覧

	マルチバイト文字	「!」,「'」を同時使用	「template1」	「'」	「"」	「/」	「¥」	「=」	「:」	「?」
ホスト名	×	×								
データベース名	×	×	×	×	×	×	×	×	×	×
ユーザ名	×	×		×	×					
パスワード名	×	×								

×…禁則文字として扱われる文字・文字列

3.3　インストール要件の確認

3.3.1　データベースサーバー

Transparent Data Encryption for PostgreSQL をインストールする PostgreSQL がインストールされているサーバーのハードウェアとソフトウェア要件について説明します。

3.3.1.1　ハードウェア要件

Transparent Data Encryption for PostgreSQL のインストールには下記のハードウェア要件を満たす必要があります。

プロセッサ	x86_64 プロセッサ
メモリ容量	約 200M バイト以上を推奨
ディスク容量	任意のディスクに約 100M バイト以上の空き領域

ヒント

AES-NI の利用

AES による暗号化および復号の高速化を目的とした CPU の命令セット AES-NI を利用するためには、以下の条件を満たす必要があります。

- PostgreSQL 9.5 以上に対して透過的暗号化機能が有効となっていること
- Windows では Transparent Data Encryption for PostgreSQL V1.2.0 以降が利用されていること
- OpenSSL がインストールされていること
    - コミュニティ推奨 Windows インストーラによりインストールされた PostgreSQL を利用します。（同梱されている OpenSSL を利用するため）

3.3.1.2　ソフトウェア要件

Transparent Data Encryption for PostgreSQL のインストールには下記のソフトウェア要件を満たす必要があります。

PostgreSQL バージョン	オペレーティングシステム（Windows）		
	Windows Server 2012（R2 を含む）	Windows Server 2016	Windows Server 2019
9.5	○	○	×
9.6	○	○	×
10	○	○	×
11	○	○	○
必要パッケージ（Windows）	Microsoft Visual C++ 2013 Redistributable (x64)		

重要

Windows プラットフォームではコミュニティが推奨している Windows インストーラーからインストールされた PostgreSQL のみをサポートします。https://www.postgresql.org/download/windows/

第 4 章
新規セットアップ

本章では、Transparent Data Encryption for PostgreSQL Enterprise Edition を初めてセットアップする手順について説明します。また、「4.6　ストリーミングレプリケーション構成への新規セットアップ （14 ページ）」手順についても説明します。

重要

Windows 版 Transparent Data Encryption for PostgreSQL は異なるバージョンを同一の端末に構成することをサポートしません。複数のバージョンを構成したい場合は Linux 版 Transparent Data Encryption for PostgreSQL のご利用をご検討ください。ただし、Linux 版 Transparent Data Encryption for PostgreSQL でも同一データベースインスタンス(データベースクラスタ)内で異なるバージョンの Transparent Data Encryption for PostgreSQL を構成することはサポートしていません。以下に Windows 版 Transparent Data Encryption for PostgreSQL のサポートしない構成例を示します。

- PostgreSQL 9.5 に対応する Transparent Data Encryption for PostgreSQL V1.2.0 と PostgreSQL 9.6 に対応する Transparent Data Encryption for PostgreSQL V1.2.0 を同一端末に構成
- Transparent Data Encryption for PostgreSQL V1.2.0 と Transparent Data Encryption for PostgreSQL V1.2.1 を同一端末に構成

ヒント

鍵管理機能は PostgreSQL データベースサーバがインストールされた端末リモートコンピュータからも実行が可能です。リモートコンピュータから鍵管理機能を利用する場合、 リモートコンピュータにも Transparent Data Encryption for PostgreSQL をインストールしてください。

4.1　新規セットアップの流れ

1. 「4.2　Transparent Data Encryption for PostgreSQL のインストール（7 ページ）」
2. 「4.3　postgresql.conf の編集（9 ページ）」
3. 「4.4　透過的暗号化機能を対話型で有効化する方法（10 ページ）」
4. 「4.5　よりセキュアな運用のための設定（12 ページ）」

   ヒント

   PostgreSQL のユーザデータを暗号化するためには、上記手順完了後に以下の作業が必要です。詳細は『透過的暗号化機能 利用の手引き』をご確認ください。

5. 利用するモードの検討
   - 簡易 TDE モード
   - 標準 TDE モード

6. 利用する暗号化アルゴリズムの検討
    - aes(Rijndael-128)
    - bf (Blowfish)
7. 暗号鍵のパスフレーズの検討
8. 暗号鍵の登録
9. 暗号化データ型を含むユーザテーブルを作成
10. 暗号化データ型のユーザデータを操作（挿入/更新/削除および参照）

4.2　Transparent Data Encryption for PostgreSQL のインストール

以下の手順に従って Transparent Data Encryption for PostgreSQL をインストールしてください。

1. Administrator 権限を持つアカウントでログインします。
2. Transparent Data Encryption for PostgreSQL Enterprise Edition インストール媒体をマウントします。
3. インストール媒体の `windows\installer\PostgreSQL` **フォルダ**の配下に格納されている対応する PostgreSQL バージョンの exe ファイルをクリックします。

    PostgreSQL フォルダについて、インストールしている PostgreSQL のバージョンが OpenSSL 1.1.0 をサポートしたものかによって対象となるフォルダが変わります。

表 4-1　PostgreSQL フォルダ

PostgreSQL フォルダ	対象 PostgreSQL バージョン
PostgreSQL9.5	PostgreSQL 9.5.0 ～ 9.5.18
PostgreSQL9.5.19	PostgreSQL 9.5.19 以降
PostgreSQL9.6	PostgreSQL 9.6.0 ～ 9.6.14
PostgreSQL9.6.15	PostgreSQL 9.6.15 以降
PostgreSQL10	PostgreSQL 10.0 ～ 10.9
PostgreSQL10.10	PostgreSQL 10.10 以降
PostgreSQL11	PostgreSQL 11.0 ～ 11.4
PostgreSQL11.5	PostgreSQL 11.5 以降

exe ファイルの命名規則は以下の通りです。

TDEforPG<*PostgreSQL バージョン*>_<*Transparent Data Encryption for PostgreSQL バージョン*>.exe

- PostgreSQL バージョン

Transparent Data Encryption for PostgreSQL が対応する PostgreSQL バージョンを示します。9.5 は 95、9.6 は 96、10 は 10、11 は 11 と表示されます。

- Transparent Data Encryption for PostgreSQL バージョン

  表記形式は X_Y_Z です。X_Y はメジャーバージョン、Z はマイナーバージョンを示します

4. [**Transparent Data Encryption for PostgreSQL Enterprise Edition (PostgreSQL XX)用の InstallShield ウィザードへようこそ**]画面が表示されますので、[**次へ**]をクリックします。

5. [**インストール先のフォルダー**]画面が表示されます。変更する場合は[**変更**]をクリックしてフォルダを指定し、[**次へ**]をクリックします。

図 4-1　インストール先のフォルダー画面

6. [**プログラムをインストールする準備ができました**]画面が表示されますので、[**インストール**]をクリックしてインストールを開始します。

7. [**InstallShield ウィザードを完了しました**]画面が表示されます。[**完了**]をクリックします。

8. インストール完了後、システム変数 PATH に Transparent Data Encryption for PostgreSQL Enterprise Edition のインストールディレクトリ配下の `lib` が設定されていることを確認します。

次の例ではコマンドプロンプトより、システム変数を確認しています。また、Transparent Data Encryption for PostgreSQL Enterprise Edition は C:\Program Files\NEC\TDEforPG96 にインストールされていることとします。

```
CMD>echo %PATH%
C:\Program Files\NEC\TDEforPG96\lib;...
```

注

システム変数 PATH に設定した PostgreSQL 本体の bin は、Transparent Data Encryption for PostgreSQL の後に設定されている必要があります。Transparent Data Encryption for PostgreSQL の前に PostgreSQL 本体の bin が設定されている場合、PostgreSQL 9.5 や PostgreSQL 9.6 では AES-NI が利用されず、パフォーマンスに影響する恐れがあります。

[システム変数 PATH の推奨設定例]

```
C:\Program Files\NEC\TDEforPG96\lib;C:\Program Files\PostgreSQL\9.6\bin;...
\
```

[システム変数 PATH の非推奨な設定例]

```
C:\Program Files\PostgreSQL\9.6\bin;C:\Program Files\NEC\TDEforPG96\lib;...
\
```

4.3　postgresql.conf の編集

Transparent Data Encryption for PostgreSQL のインストール完了後、透過的暗号化機能を利用するために PostgreSQL の設定ファイル（postgresql.conf）を変更し、設定の変更を有効にします。

1. PostgreSQL の設定ファイル（postgresql.conf）の shared_preload_libraries パラメータに Transparent Data Encryption for PostgreSQL のダイナミックリンクライブラリ data_encryption.dll を設定します。

   [postgresql.conf 設定例]

   次の設定例では、C:\Program Files\NEC\TDEforPG96 に Transparent Data Encryption for PostgreSQL がインストールされていることとします。

   ```
   shared_preload_libraries = '"C:\\Program Files\\NEC\\TDEforPG96\\lib\
   \data_encryption.dll"'
   ```

2. 変更した設定を有効にするため、PostgreSQL を再起動します。

   次の例では、サービスを再起動することで PostgreSQL を再起動します。

```
CMD>sc stop postgresql-x64-9.6
CMD>sc start postgresql-x64-9.6
```

ヒント

サービスとして登録していない場合は、pg_ctl[*1] コマンドを利用して PostgreSQL を再起動しています。

```
CMD> pg_ctl restart
```

4.4　透過的暗号化機能を対話型で有効化する方法

以下の手順に従って対話型で透過的暗号化機能を有効化してください。

重要

透過的暗号化機能を有効化する際に cipher_setup.bat を介して、PowerShell のスクリプトを起動しています。そのため、Windows PowerShell スクリプトの実行ポリシーを RemoteSigned もしくは Unrestricted に設定する必要があります。設定方法は Microsoft 公式ページをご確認ください。なお、Windows Server 2012 にインストールされている PowerShell V3 は、デフォルトではスクリプトの実行が許可されていません。そのため、PowerShell スクリプトを実行すると、以下のようなエラーが出力され終了します（Windows Server 2012 R2 以降の PowerShell では、デフォルトで RemomoteSigned に変更されています）。

```
PS> Get-ExecutionPolicy
Restricted
PS> .\cipher_setup.ps1
.\cipher_setup.ps1 : このシステムではスクリプトの実行が無効になっているため、
ファイル C:\Program Files\nec\TDEforPG96\bin\cipher_setup.ps1 を読み込むことができません。
詳細については、「about_Execution_Policies」(http://go.microsoft.com/fwlink/?LinkID=135170) を参照してください。
発生場所 行:1 文字:1
+ .\cipher_setup.ps1
+ ~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : セキュリティ エラー: (: ) []、PSSecurityException
    + FullyQualifiedErrorId : UnauthorizedAccess
```

1. Administrator 権限を持つアカウントでログインします。
2. [**スタートメニュー**]>[**cipher_setup の実行**]をクリックします。
3. [**NEC TDE for PG VX.Y.Z Cipher Setup**]画面が表示されます。

*1　pg_ctl コマンドの詳細な利用方法は PostgreSQL マニュアルをご確認ください。

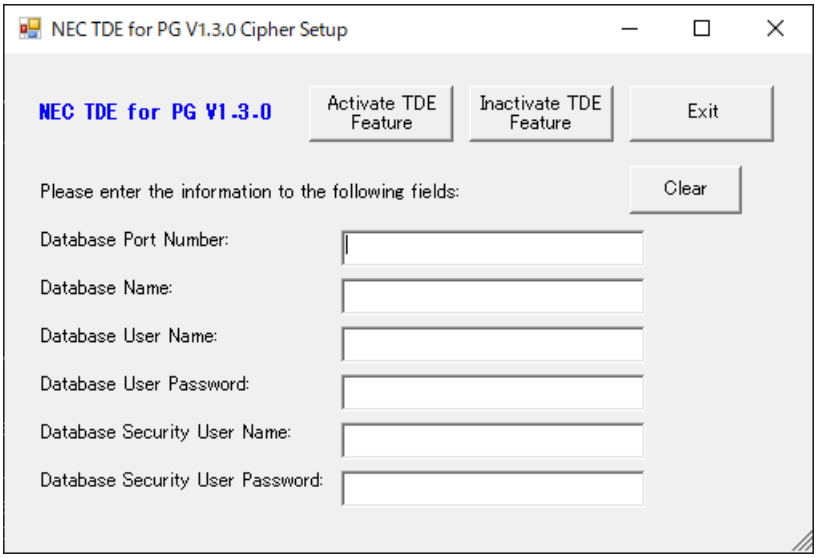

図 4-2　NEC TDE for PG VX.Y.Z Cipher Setup 画面

注

GUI の右上の[×]ボタンをクリックすると、警告なしに閉じられます。

表 4-2　各項目の説明

項目	説明	有効化の際の入力有無
Database Port Number	ポート番号	入力必要
Database Name	データベース名	入力必要
Database User Name	スーパユーザ名	入力必要
Database User Password	スーパユーザのパスワード	入力必要
Database Security User Name	セキュリティ管理ユーザ名	入力必要
Database Security User Password	セキュリティ管理ユーザのパスワード	入力必要

4. 「3.2　透過的暗号化機能をセットアップするために必要な情報（3 ページ）」を参考に PostgreSQL への接続情報、およびセキュリティ管理ユーザを入力し、[**Activate TDE Feature**]をクリックします。

   透過的暗号化機能を有効化するデータベース は事前に作成されている必要があります。入力したセキュリティ管理ユーザ名が PostgreSQL に存在しない場合、新規に PosgreSQL ユーザを作成します。

   ## 注

   セキュリティ管理ユーザとして PostgreSQL のスーパユーザを指定することはできません。

5. [**Activate confirm**]が表示されます。[**はい**]をクリックします。

入力した情報に問題が無ければ[**Activate success**]ダイアログが表示されます。表示されたメッセージを確認し、[**OK**]をクリックします。これで指定したデータベースに対して透過的暗号化機能が有効化されます。また、セキュリティ管理ユーザの接続情報が記載された設定ファイルが作成されます（本手順では `C:\Program Files\NEC\TDEforPG96\conf\pgtde_secuser.properties`）。暗号鍵を管理するオペレーティングシステムユーザは、

このファイルを透過的暗号化機能コマンド（pgtde）実行時の接続情報ファイルとして使用することが可能です。

4.5　よりセキュアな運用のための設定

透過的暗号化機能は、OS ユーザおよびファイルの権限を適切に設定することでよりセキュアな運用が実現できます。よりセキュアな運用を行いたい場合は以下の設定を実施してください。

1. 透過的暗号化機能をよりセキュアな状態で運用するためには、各機能毎に OS ユーザおよび OS グループを作成します。

   それぞれの OS ユーザが適切な PostgreSQL ユーザを使用するような運用方針を策定する必要があります。作成するユーザと対応する PostgreSQL ユーザの一覧については下記をご参考の上作成してください。

表 4-3　作成する OS ユーザ一覧

OS ユーザ	OS グループ	役割	使用可能な PostgreSQL ユーザ
データベース管理者	透過的暗号化機能管理グループ	PostgreSQL 起動ユーザであり、PostgreSQL に対する全権限を持つユーザ。	スーパユーザ
セキュリティ管理者	透過的暗号化機能管理グループ	透過的暗号化機能で利用する鍵の管理権限を持つユーザ	透過的暗号化機能のセットアップで作成または指定したセキュリティ管理ユーザ
アプリケーション管理者（アプリケーション開発者）	透過的暗号化機能利用グループ	透過的暗号化機能を利用しているユーザデータに対する暗号化・復号権限を持つユーザ	透過的暗号化機能を利用するユーザデータにアクセスできる一般ユーザ

   次の例では透過的暗号化機能管理グループ「tde_manager」と透過的暗号化機能利用グループ「tde_user」を作成し、セキュリティ管理者「secuser」、アプリケーション管理者（アプリケーション開発者）「apuser」をそれぞれのグループに所属させるよう作成しています。

```
CMD>NET USER secuser /ADD ******
CMD>NET USER apuser /ADD ******
CMD>NET LOCALGROUP tde_manager /ADD
CMD>NET LOCALGROUP tde_user /ADD
CMD>NET LOCALGROUP tde_manager secuser /ADD
CMD>NET LOCALGROUP tde_user apuser /ADD
CMD>NET LOCALGROUP Users secuser /ADD
CMD>NET LOCALGROUP Users apuser /ADD
```

2. 透過的暗号化機能をよりセキュアな状態で運用するためには、各種ファイルをそれぞれ適切な所有者に設定します。

   次の表を参考に、作成したユーザ毎にファイルの権限を設定してください。

表 4-4　アクセス権限設定を推奨する透過的暗号化機能関連ファイル一覧

対象ファイル	所有者
conf/pgtde_secuser.properties	セキュリティ管理者

対象ファイル	所有者
lib/jar/pgtde.jar	アプリケーション管理者（アプリケーション開発者）
lib/jar/pgtde_regist.jar	セキュリティ管理者

次の例では、セキュリティ管理者に「secuser」、アプリケーション管理者（アプリケーション開発者）に「apuser」として各種ファイルの所有者を設定しています。また、PostgreSQL 9.6 用の Transparent Data Encryption for PostgreSQL が C:\Program Files\NEC\TDEforPG96 にインストールされていることとします。

```
CMD>CD "C:\Program Files\NEC\TDEforPG96"
CMD>ICACLS "conf\pgtde_secuser.properties" /grant secuser:F
CMD>ICACLS "conf\pgtde_secuser.properties" /inheritance:r
CMD>ICACLS "lib\jar\pgtde.jar" /grant apuser:F
CMD>ICACLS "lib\jar\pgtde.jar" /grant Administrators:F
CMD>ICACLS "lib\jar\pgtde.jar" /inheritance:r
CMD>ICACLS "lib\jar\pgtde_regist.jar" /grant secuser:F
CMD>ICACLS "lib\jar\pgtde_regist.jar" /grant Administrators:F
CMD>ICACLS "lib\jar\pgtde_regist.jar" /inheritance:r
```

上記ファイルの権限設定により、透過的暗号化機能コマンド（pgtde）の各-m オプションの実行がユーザ毎に以下のように制限されます。(各-m オプションの詳細は『透過的暗号化機能 利用の手引』をご確認ください)

表 4-5　モード毎実行可能ユーザ一覧

各-m オプション	実行可能ユーザ
暗号鍵の登録・更新(-m regist)	セキュリティ管理者
モードの変更(-m switch)	
利用状況を表示(-m show)	
最新の暗号鍵による再暗号化(-m cipher)	アプリケーション管理者（アプリケーション開発者）

3. 透過的暗号化機能を利用したいデータベースの一般ユーザは、暗号鍵情報テーブルに対して適切なアクセス権限を設定します。対象のデータベースに存在する暗号鍵情報テーブル(cipher_key_table)に対して GRANT 文を利用して一般ユーザに UPDATE と DELETE 権限を設定します。

   次の例では、データベースの一般ユーザ「apuser」に対して暗号鍵情報テーブル(cipher_key_table)の UPDATE と DELETE 権限を設定しています。

```
=# CREATE ROLE apuser WITH LOGIN ENCRYPTED PASSWORD '******';
=# GRANT UPDATE ON cipher_key_table TO apuser;
=# GRANT DELETE ON cipher_key_table TO apuser;
```

ヒント

PostgreSQL のセキュリティ管理ユーザに透過的暗号化機能のセットアップで作成したユーザ以外の一般ユーザを割り当てる場合、対象のデータベースに対して次の権限を設定しま

す。次の例では 一般ユーザ「secuser」を透過的暗号化機能のセキュリティ管理者用として設定しています。

```
=# CREATE ROLE secuser WITH LOGIN ENCRYPTED PASSWORD '******';
=# GRANT INSERT ON cipher_key_table TO secuser;
=# GRANT UPDATE ON cipher_key_table TO secuser;
=# GRANT DELETE ON cipher_key_table TO secuser;
=# GRANT EXECUTE ON FUNCTION cipher_key_backup() TO secuser;
```

4.6　ストリーミングレプリケーション構成への新規セットアップ

PostgreSQL の標準機能である、ストリーミングレプリケーションを利用した環境に Transparent Data Encryption for PostgreSQL を初めてインストールする手順について説明します。以下の手順でセットアップを行います。

表 4-6　インストール時の手順要否

手順	作業項目		参照先
	プライマリサーバ	スタンバイサーバ	
1	PostgreSQL のインストール		関連リンク参照
2	インスタンスの作成・設定		関連リンク参照
3		インスタンスの作成・設定	関連リンク参照
4	ストリーミングレプリケーションの状態確認		関連リンク参照
5	Transparent Data Encryption for PostgreSQL のインストール		「4.6.1　Transparent Data Encryption for PostgreSQL のインストール（手順 5）（15 ページ）」
6	postgresql.conf の編集		「4.6.2　postgresql.conf の編集（手順 6）（15 ページ）」
7	透過的暗号化機能の有効化		「4.6.3　透過的暗号化機能の有効化（手順 7）（15 ページ）」
8	よりセキュアな運用のための設定		「4.6.4　よりセキュアな運用のための設定（手順 8）（15 ページ）」

関連リンク

PostgreSQL のインストール（PostgreSQL の Windows インストーラ、Linux ディストリビューション・パッケージなどのリンク集、およびインストールガイド URL https://www.postgresql.jp/download）

インスタンスの作成・設定（最新バージョンの PostgreSQL マニュアル URL https://www.postgresql.jp/document/current/html/admin.html）

ストリーミングレプリケーションの状態確認（最新バージョンの PostgreSQL マニュアル URL https://www.postgresql.jp/document/current/html/high-availability.html）

4.6.1　Transparent Data Encryption for PostgreSQL のインストール（手順 5）

ストリーミングレプリケーションを利用する場合は、「4.2　Transparent Data Encryption for PostgreSQL のインストール（7 ページ）」を参考にプライマリサーバとスタンバイサーバの両方にインストールを行ってください。

重要

インストールパスの指定は、プライマリサーバとスタンバイサーバを同じ フォルダパスに統一する必要があります。

4.6.2　postgresql.conf の編集（手順 6）

ストリーミングレプリケーションを利用する場合は、「4.3　postgresql.conf の編集（9 ページ）」を参考にプライマリサーバとスタンバイサーバの両方の postgresql.conf の shared_preload_libraries パラメータに Transparent Data Encryption for PostgreSQL のダイナミックリンクライブラリ data_encryption.dll を設定してください。

4.6.3　透過的暗号化機能の有効化（手順 7）

「4.4　透過的暗号化機能を対話型で有効化する方法（10 ページ）」を参考にプライマリサーバのみ透過的暗号化機能を有効化してください。

4.6.4　よりセキュアな運用のための設定（手順 8）

ストリーミングレプリケーションを利用した環境でよりセキュアな運用を行いたい場合は、「4.5　よりセキュアな運用のための設定（12 ページ）」を参考にプライマリサーバとスタンバイサーバの両方を同一の構成となるよう設定してください。

第 5 章
アンインストール

本章では、Transparent Data Encryption for PostgreSQL Enterprise Edition をアンインストールする手順について説明します。また、「5.6　ストリーミングレプリケーション構成からのアンインストール（20 ページ）」や「5.7　透過的暗号化機能を対話型で再有効化する方法（20 ページ）」についても説明します。

重要

透過的暗号化機能を無効化しても暗号化されたデータは復号されません。そのため、Transparent Data Encryption for PostgreSQL Enterprise Edition アンインストール後も暗号化されたデータを利用する場合は、アンインストール前に暗号化されたデータを復号してください。

5.1　アンインストールの流れ

1. 「5.2　透過的暗号化機能を対話型で無効化する方法（16 ページ）」
2. 「5.3　Transparent Data Encryption for PostgreSQL のアンインストール（17 ページ）」
3. 「5.4　postgresql.conf の編集（19 ページ）」
4. 「5.5　インストールディレクトリの削除（20 ページ）」

5.2　透過的暗号化機能を対話型で無効化する方法

以下の手順に従って対話型で透過的暗号化機能を無効化してください。

1. Administrator 権限を持つアカウントでログインします。
2. [**スタートメニュー**]>[**cipher_setup の実行**]をクリックします。
3. [**NEC TDE for PG VX.Y.Z Cipher Setup**]画面が表示されます。

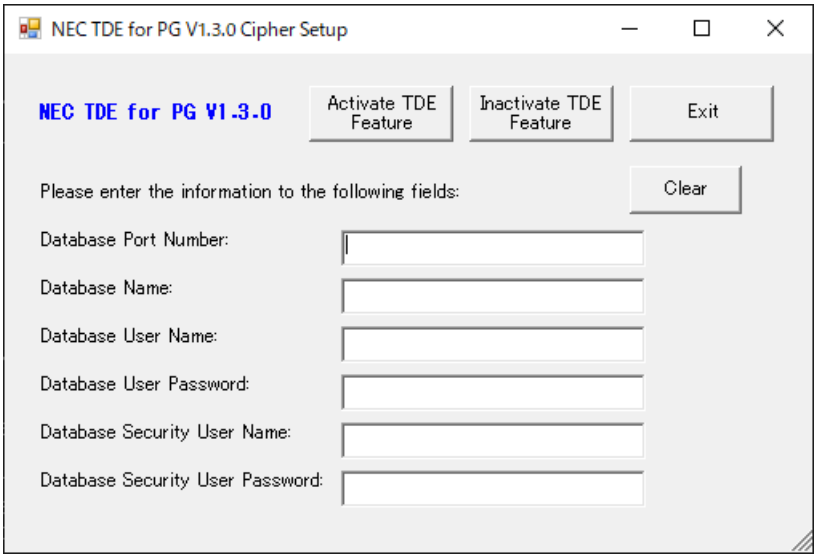

図 5-1　NEC TDE for PG VX.Y.Z Cipher Setup 画面

注

GUI の右上の[**×**]ボタンをクリックすると、警告なしに閉じられます。

表 5-1　各項目の説明

項目	説明	無効化の際の入力有無
Database Port Number	ポート番号	入力必要
Database Name	データベース名	入力必要
Database User Name	スーパユーザ名	入力必要
Database User Password	スーパユーザのパスワード	入力必要
Database Security User Name	セキュリティ管理ユーザ名	入力不要
Database Security User Password	セキュリティ管理ユーザのパスワード	入力不要

4. PostgreSQL への接続情報、およびセキュリティ管理ユーザを入力し、[**Inactivate TDE Feature**]をクリックします。
5. [**Inactivate confirm**]ダイアログが表示されますので、[**はい**]をクリックします。

入力した情報に問題が無ければ[**Inactivate success!**]ダイアログが表示されます。表示されたメッセージを確認し、[**OK**]をクリックします。これで指定したデータベースに対して透過的暗号化機能が無効化されます。

5.3　Transparent Data Encryption for PostgreSQL のアンインストール

以下の手順に従って Transparent Data Encryption for PostgreSQL をアンインストールしてください。

1. Administrator 権限を持つアカウントでログインします。

2. [**コントロールパネル**]>[**プログラムと機能**]を選択し、[**プログラムと機能**]画面を起動します。

3. [**Transparent Data Encryption for PostgreSQL Enterprise Edition (PostgreSQL *XX*)**]を右クリックし、[**アンインストール**]をクリックします。

   XX は PostgreSQL のメジャーバージョンです。

4. [**プログラムと機能**]ダイアログが起動し、アンインストールを実行するか確認されるので[**はい**]を選択します。[**いいえ**]を選択した場合、アンインストールは中止されます。

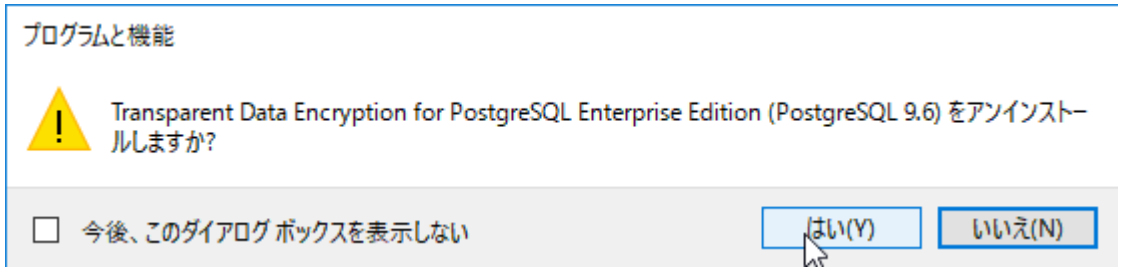

図 5-2　プログラムと機能ダイアログ

5. アンインストールの前に透過的暗号化機能を無効化したか確認するダイアログが表示されるので[**はい**]を選択します。[**いいえ**]を選択した場合、アンインストールは中止されます。

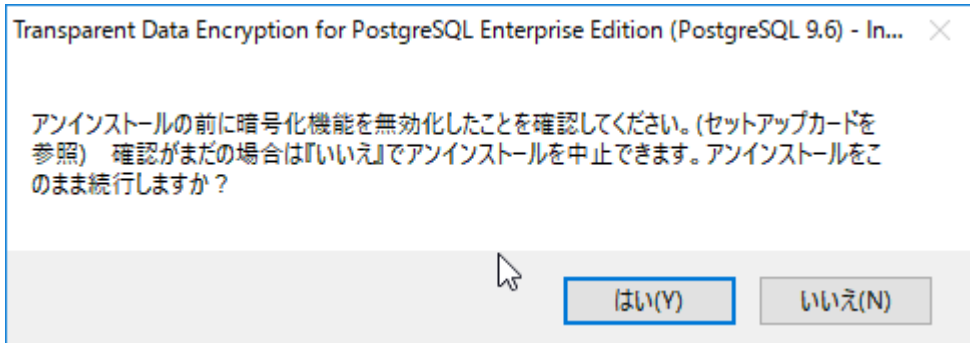

図 5-3　透過的暗号化機能の無効化する際の確認ダイアログ

6. 「アンインストールを続行します」と確認するダイアログが表示されるので[**OK**]をクリックします。

7. セットアップを完了するためには再起動が必要なことを通知するダイアログが表示されるので[**OK**]をクリックします。

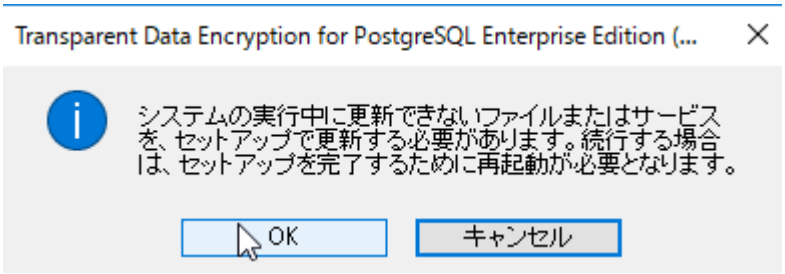

図 5-4　再起動を促すダイアログ

8. 必要に応じて、再起動します。

5.4　postgresql.conf の編集

Transparent Data Encryption for PostgreSQL のインストール完了後、透過的暗号化機能を利用停止するために PostgreSQL の設定ファイル（`postgresql.conf`）を変更し、設定の変更を有効にします。

1. PostgreSQL の設定ファイル（`postgresql.conf`）の shared_preload_libraries パラメータに Transparent Data Encryption for PostgreSQL のダイナミックリンクライブラリ `data_encryption.dll` を削除、またはパラメータ自体をコメントアウトします。

   [postgresql.conf 設定例]

```
shared_preload_libraries = ''
```

2. 変更した設定を有効にするため、PostgreSQL を再起動します。

   次の例では、サービスを再起動することで PostgreSQL を再起動しています。

```
CMD>sc stop postgresql-x64-9.6
CMD>sc start postgresql-x64-9.6
```

ヒント

サービスとして登録していない場合は、`pg_ctl`[*1] コマンドを利用して PostgreSQL を再起動しています。

```
CMD> pg_ctl restart
```

*1　`pg_ctl` コマンドの詳細な利用方法は PostgreSQL マニュアルをご確認ください。

5.5　インストールディレクトリの削除

Transparent Data Encryption for PostgreSQL を今後利用しない場合、インストールディレクトリを削除します。

1. インストールディレクトリを削除します。

   次の例では、PostgreSQL 9.6 用の Transparent Data Encryption for PostgreSQL が `C:\Program Files\NEC\TDEforPG96` にインストールされていることとします。

```
CMD>CD "C:\Program Files\NEC"
CMD>dir
2018/03/15  11:24    <DIR>          TDEforPG96
CMD>RMDIR /S tdeforpg96
TDEforPG96、よろしいですか (Y/N)?Y
```

5.6　ストリーミングレプリケーション構成からのアンインストール

ストリーミングレプリケーションを利用した環境から Transparent Data Encryption for PostgreSQL をアンインストールする手順について説明します。以下の手順でアンインストールします。

また、アンインストール時のプライマリサーバとスタンバイサーバのセットアップ手順の要否については、以下の通りです。

表 5-2　アンインストール時の手順要否

手順	作業項目		参照先
	プライマリサーバ	スタンバイサーバ	
1	透過的暗号化機能の無効化		「5.2　透過的暗号化機能を対話型で無効化する方法（16 ページ）」
2	Transparent Data Encryption for PostgreSQL のアンインストール		「5.3　Transparent Data Encryption for PostgreSQL のアンインストール（17 ページ）」
3	postgresql.conf の編集		「5.4　postgresql.conf の編集（19 ページ）」
4	インストールディレクトリの削除		「5.5　インストールディレクトリの削除（20 ページ）」

5.7　透過的暗号化機能を対話型で再有効化する方法

以下の手順に従って対話型で透過的暗号化機能を再有効化してください。

注

透過的暗号化機能を有効化している状態でデータベースを削除した場合、同名のデータベースを再作成しても透過的暗号化機能を有効化することはできません。再度有効化したい場合、PP サポートサービスにご連絡ください。

1. Administrator 権限を持つアカウントでログインします。
2. [**スタートメニュー**]>[**cipher_setup の実行**]をクリックします。
3. [**NEC TDE for PG VX.Y.Z Cipher Setup**]画面が表示されます。

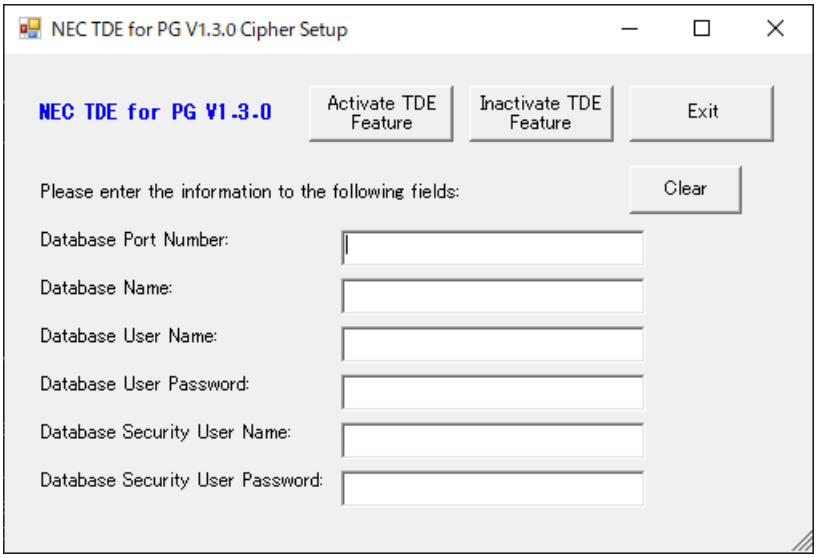

図 5-5　NEC TDE for PG VX.Y.Z Cipher Setup 画面

注

GUI の右上の[**×**]ボタンをクリックすると、警告なしに閉じられます。

表 5-3　各項目の説明

項目	説明
Database Port Number	ポート番号
Database Name	データベース名
Database User Name	スーパユーザ名
Database User Password	スーパユーザのパスワード
Database Security User Name	セキュリティ管理ユーザ名
Database Security User Password	セキュリティ管理ユーザのパスワード

4. 「3.2　透過的暗号化機能をセットアップするために必要な情報（3 ページ）」を参考に PostgreSQL への接続情報、およびセキュリティ管理ユーザを入力し、[**Activate TDE Feature**]をクリックします。

   透過的暗号化機能を有効化するデータベース は事前に作成されている必要があります。入力したセキュリティ管理ユーザ名が PostgreSQL に存在しない場合、新規に PosgreSQL ユーザを作成します。

注

セキュリティ管理ユーザとして PostgreSQL のスーパユーザを指定することはできません。

5. [**Activate confirm**]が表示されます。[**はい**]をクリックします。

入力した情報に問題が無ければ[**Activate success**]ダイアログが表示されます。表示されたメッセージを確認し、[**OK**]をクリックします。これで指定したデータベースに対して透過的暗号化機能が再有効化されます。また、セキュリティ管理ユーザの接続情報が記載された設定ファイルが作成されます（本手順では `C:\Program Files\NEC\TDEforPG96\conf\pgtde_secuser.properties`）。暗号鍵を管理する OS ユーザは、このファイルを透過的暗号化機能コマンド（`pgtde`）実行時の接続情報ファイルとして使用することが可能です。

第 6 章
アップグレード

本章では下記 3 パターンのアップグレードについて説明します。

- 「6.1　Free Edition から Enterprise Edition へのアップグレード（23 ページ）」
- 「6.2　Transparent Data Encryption for PostgreSQL のアップグレード（25 ページ）」
- 「6.3　透過的暗号化機能が有効な PostgreSQL のメジャーバージョンアップグレード（27 ページ）」

注

以下のような場合は PP サポートサービスにご連絡ください。

- クラスタ構成のアップグレードをご検討の場合

  クラスタ構成の仕様（利用製品）によっては、待機系のアップグレード手順が異なります。
- クラスタ構成で透過的暗号化機能を有効化した端末以外で透過的暗号化機能を制御したい場合
- PostgreSQL の標準機能ストリーミングレプリケーション構成でのアップグレードをご検討の場合

6.1　Free Edition から Enterprise Edition へのアップグレード

アップグレードを行う前に `pg_dumpall` を使用してデータベース全体のバックアップを取得していただくことを推奨いたします。

Transparent Data Encryption for PostgreSQL Free Edition を利用しているデータベースの Transparent Data Encryption for PostgreSQL Enterprise Edition へのアップグレードを行う場合、バージョン番号 X.Y.Z.N の X と Y がアップグレード先のバージョンと同一である場合に限り、以下の手順に従ってアップグレードを行ってください。この条件に合致しない場合は、後述する「6.3　透過的暗号化機能が有効な PostgreSQL のメジャーバージョンアップグレード（27 ページ）」をご参照ください。

1. Administrator 権限を持つアカウントでログインします。
2. Transparent Data Encryption for PostgreSQL Free Edition が提供している `cipher_setup.bat` を使用し、データベースにインストールされている透過的暗号化機能を無効化してください。

   Transparent Data Encryption for PostgreSQL Free Edition では `cipher_setup.bat` を実行した際に表示される画面が Transparent Data Encryption for PostgreSQL Enterprise Edition と異なり、以下のように表示されます。

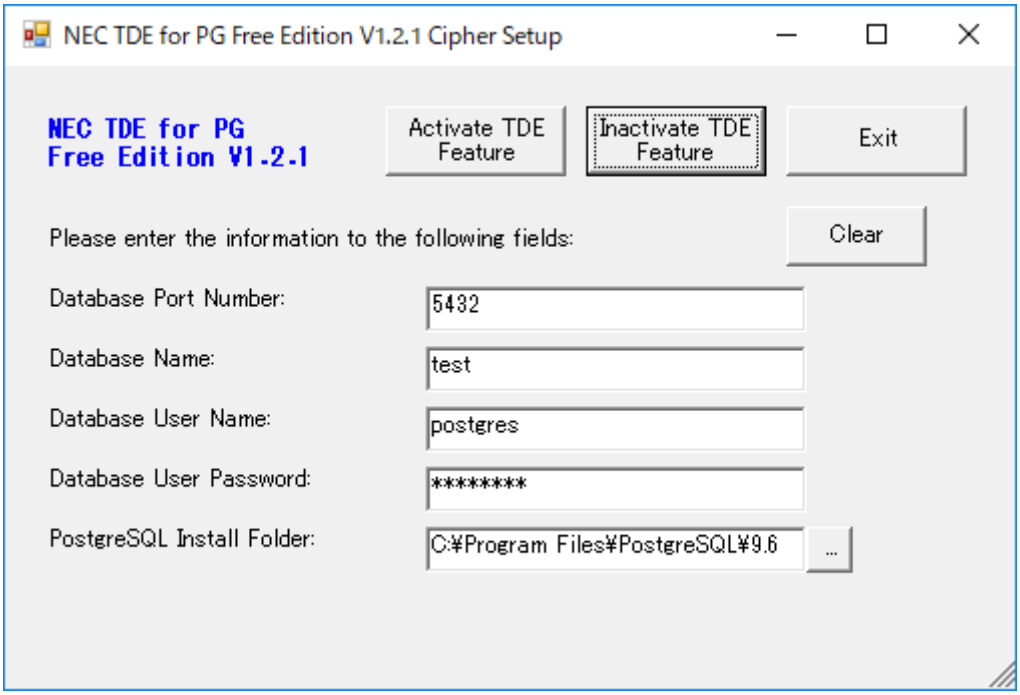

図 6-1　NEC TDE for PG Free Edition VX.Y.Z Cipher Setup 画面

3. 透過的暗号化機能の無効化が完了したデータベースにスーパユーザで接続し、イ ンストールされている pgcrypto モジュールを DROP EXTENSION クエリでアンイ ンストールします。

```
=# DROP EXTENSION pgcrypto;
DROP EXTENSION
```

4. 透過的暗号化機能を利用しているデータベースを停止します。

   次の例では、サービスを停止することで PostgreSQL を停止します。

```
CMD>sc stop postgresql-x64-9.6
```

ヒント

サービスとして登録していない場合は、`pg_ctl`[*1] コマンドを利用して PostgreSQL を停止します。

```
CMD> pg_ctl stop
```

5. 「4.2　Transparent Data Encryption for PostgreSQL のインストール（7 ページ）」を参考にアップグレード先の Transparent Data Encryption for PostgreSQL Enterprise Edition をインストールします。

*1　`pg_ctl` コマンドの詳細な利用方法は PostgreSQL マニュアルをご確認ください。

6. 「4.3　postgresql.conf の編集（9 ページ）」を参考に `postgresql.conf` ファイルを編集し、PostgreSQL を起動します。shared_preload_libraries パラメータには、Transparent Data Encryption for PostgreSQL Free Edition で設定した値が記載されているため、当該設定を削除の上で設定値を変更してください。

7. 新規インストールした Transparent Data Encryption for PostgreSQL Enterprise Edition の `cipher_setup.bat` を使用して透過的暗号化機能を再有効化することでアップグレードが完了します。再有効化の手順は「5.7　透過的暗号化機能を対話型で再有効化する方法（20 ページ）」をご確認ください。

   Transparent Data Encryption for PostgreSQL Free Edition からのアップグレードを伴う対話型の再有効化の場合、[**Activate confirm**]画面（アップグレード確認画面）が表示されますので、問題がない場合は[**Yes**]をクリックします。次の例 では Transparent Data Encryption for PostgreSQL Free Edition V1.2.1 から Transparent Data Encryption for PostgreSQL Enterprise Edition V1.3.0 へのアップグレー ドを実施しています。

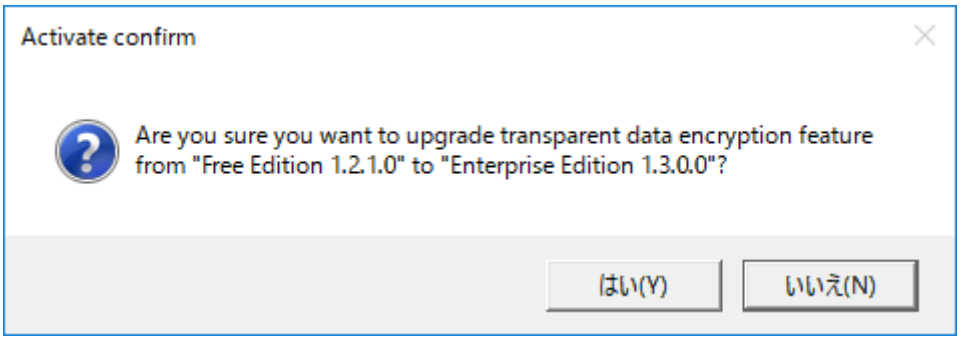

図 6-2　Activate confirm 画面

8. 以上でアップグレードは完了となります。必要に応じて旧バージョンの Transparent Data Encryption for PostgreSQL Free Edition をアンインストールしてください。

6.2　Transparent Data Encryption for PostgreSQL のアップグレード

アップグレードを行う前に `pg_dumpall` を使用してデータベース全体のバックアップを取得していただくことを推奨いたします。

Transparent Data Encryption for PostgreSQL のメジャーバージョン、マイナーバージョンともに以下の手順に従ってアップグレードを行ってください。

1. Administrator 権限を持つアカウントでログインします。

2. 「5.2　透過的暗号化機能を対話型で無効化する方法（16 ページ）」を参考に透過的暗号化機能を無効化します。

3. 透過的暗号化機能を利用しているデータベースを停止します。

   次の例では、サービスを停止することで PostgreSQL を停止します。

```
CMD>sc stop postgresql-x64-9.6
```

ヒント

サービスとして登録していない場合は、pg_ctl[*2] コマンドを利用して PostgreSQL を停止します。

```
CMD> pg_ctl stop
```

4. 「5.3　Transparent Data Encryption for PostgreSQL のアンインストール（17 ページ）」を参考に Transparent Data Encryption for PostgreSQL をアンインストールします。
5. 「5.4　postgresql.conf の編集（19 ページ）」を参考に PostgreSQL の設定ファイル（postgresql.conf）の shared_preload_libraries パラメータに Transparent Data Encryption for PostgreSQL のダイナミックリンクライブラリ data_encryption.dll を削除、またはパラメータ自体をコメントアウトします。
6. インストール媒体の windows\installer フォルダ配下に格納されている対応する PostgreSQL バージョンの exe ファイルをクリックします。

   exe ファイルの命名規則は以下の通りです。

表 6-1　exe ファイルの命名規則

Edition	命名規則
Enterprise Edition	TDEforPG<*PostgreSQL バージョン*>_<*Transparent Data Encryption for PostgreSQL バージョン*>.exe

   - PostgreSQL バージョン

     Transparent Data Encryption for PostgreSQL が対応する PostgreSQL バージョンを示します。9.5 は 95、9.6 は 96、10 は 10、11 は 11 と表示されます。

   - Transparent Data Encryption for PostgreSQL バージョン

     表記形式は X_Y_Z です。X_Y はメジャーバージョン、Z はマイナーバージョンを示します

7. 「4.2　Transparent Data Encryption for PostgreSQL のインストール（7 ページ）」を参考に Transparent Data Encryption for PostgreSQL をインストールします。

注

インストール先ディレクトリを指定してインストールした場合

Transparent Data Encryption for PostgreSQL の旧バージョンを任意のフォルダにインストールしている場合は、同一のフォルダをインストール先フォルダとして指定します。指定せずに再インストールした場合、デフォルトで指定されたフォルダにインストールされます。

*2　pg_ctl コマンドの詳細な利用方法は PostgreSQL マニュアルをご確認ください。

8. 透過的暗号化機能を利用しているデータベースを起動します。

```
CMD> sc start postgresql-x64-9.6
```

ヒント

サービスとして登録していない場合は、pg_ctl コマンドを利用して PostgreSQL を起動します。

```
CMD> pg_ctl start
```

9. 「5.7　透過的暗号化機能を対話型で再有効化する方法（20 ページ）」を参考に透過的暗号化機能を再有効します。

アップグレードを伴う対話型の再有効化の場合、[**Activate confirm**]画面（アップグレード確認画面）が表示されますので、問題がない場合は[**Yes**]をクリックします。次の例では V1.2.1 から V1.3.0 へのアップグレードを実施しています。

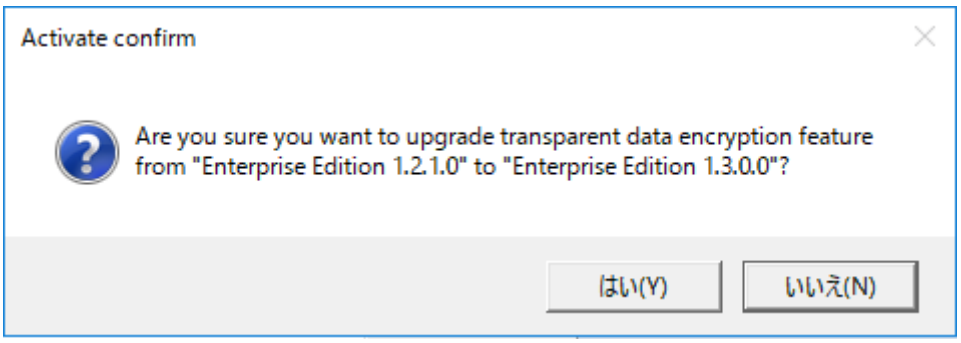

図 6-3　Activate confirm 画面

入力した情報に問題が無ければ[**Activate success**]ダイアログが表示されます。表示されたメッセージを確認し、[**OK**]をクリックします。これで指定したデータベースに対して透過的暗号化機能が有効化されます。また、セキュリティ管理ユーザの接続情報が記載された設定ファイルが作成されます（本手順では C:\Program Files\NEC\TDEforPG96\conf\pgtde_secuser.properties）。暗号鍵を管理するオペレーティングシステムユーザは、このファイルを透過的暗号化機能コマンド（pgtde）実行時の接続情報ファイルとして使用することが可能です。

10. 旧バージョンでよりセキュアな運用のための設定を行っていた場合、再度「4.5　よりセキュアな運用のための設定（12 ページ）」を参考に設定を行います。

6.3　透過的暗号化機能が有効な PostgreSQL のメジャーバージョンアップグレード

アップグレードを行う前に pg_dumpall を使用してデータベース全体のバックアップを取得していただくことを推奨いたします。

透過的暗号化機能が有効となっている PostgreSQL のメジャーバージョンアップグレードを行う場合、以下の手順に従って PostgreSQL のメジャーバージョンアップグレードを行ってください。透過的暗号化機能を利用しているデータベースは PostgreSQL の標準機能である pg_dump/pg_restore コマンドを利用します。透過的暗号化機能を利用するデータベースを含んだ状態で pg_dumpall コマンドや pg_upgrade コマンドを使ってデータベースクラスタ全体を移行する方法はサポートしていません。本節の手順を利用することで、Transparent Data Encryption for PostgreSQL と PostgreSQL のメジャーバージョンアップグレードを同時に行うことできます。本節の例では、Transparent Data Encryption for PostgreSQL V1.3.0 がセットアップされた PostgreSQL 9.6 を PostgreSQL 11 にアップグレードします。また、手順では Transparent Data Encryption for PostgreSQL が C:\Program Files\NEC（デフォルト）にインストールされていることとし、透過的暗号化機能コマンド（pgtde）のデータベース接続ファイルとして C:\Program Files\NEC\TDEforPG96\conf\pgtde_secuser.properties を使用します。

注

PostgreSQL 9.6.18 まで動作確認を行っておりますが、それ以降の PostgreSQL バージョンにて手順が失敗する場合は別途お問合せください。

1. 透過的暗号化機能を利用しているデータベースが簡易 TDE モードを利用している場合は、pgtde -m switch --standard-tde[*3] コマンドを利用して標準 TDE モードに変更します。

```
CMD> "C:\Program Files\NEC\TDEforPG96\bin\pgtde.bat" -m switch --standard-tde
 ^
More? -conf "C:\Program Files\NEC\TDEforPG96\conf\pgtde_secuser.properties"
Enter current data key: ******
Are you sure you want to switch key management mode of "tdedb"(DATABASE) to "S
tandard TDE mode" ? (Press Y(y) key to execute): Y
New key version 1 is registered to tdedb

Switch key management mode to standard TDE mode is successfully.
```

2. 対象のデータベースが複数の暗号鍵を利用している場合、次の手順でユーザデータを再暗号化します。

   注

   最新の暗号鍵によるデータ再暗号化は負荷状況に注意する必要があります。

   a. 対象のデータベースに接続し、次のクエリを発行した結果が「1」でなければ次の手順を実施します。

*3 モードの切り替え手順や pgtde コマンドの詳細は対象バージョンの『透過的暗号化機能 利用の手引き』をご確認ください。

```
=# SELECT count(*) FROM public.cipher_key_table;
```

b. pgtde -m cipher --reset コマンドを使用してデータの再暗号化 を実施します。

```
CMD> "C:\Program Files\NEC\TDEforPG96\bin\pgtde.bat" -m cipher --reset ^
More? -conf "C:\Program Files\NEC\TDEforPG96\conf\pgtde_secuser.properti
es"
Enter current data key:******
Are you sure you want to Reencrypt "tdedb"(DATABASE) with Interval="0" an
d Reset="true"? (Press Y(y) key to execute): Y
All data in tdedb are reencrypted
```

3. アップグレード対象のデータベースに対して pg_dump[*4] コマンドを実行します。透過的暗号化機能を利用するデータベースが複数存在する場合はデータベース毎に実施してください。なお、pg_dump 実行前に PGOPTIONS 環境変数で encrypt.enable パラメータを off にすることでユーザデータを暗号化したままの状態でバックアップすることが可能です。

   次の例では、ダンプファイル名として「pg_dump_tdedb.dump」、バックアップ対象のデータベースとして「tdedb」を指定しています。

```
CMD> set PGOPTIONS=-c encrypt.enable=off
CMD> pg_dump -f pg_dump_tdedb.dump -Fc tdedb
```

4. ダンプファイルには旧バージョンの透過的暗号化機能オブジェクトも含まれているため、当該ダンプファイルをそのまま利用してデータベースのリストアを行うことはできません。そのため、旧バージョンの透過的暗号化機能オブジェクトを除いたユーザデータのみをリストアする為のリストアリストを作成する必要があります。以下の手順に従って透過的暗号化機能を利用するデータベース毎のバックアップファイルからリストア対象ユーザデータオブジェクトファイルを作成してください。

   a. バックアップしたダンプファイル(本節の手順では pg_dump_tdedb.dump)のリストファイルを作成します。

      次の例では、ダンプファイルとして前の手順で作成した「pg_dump_tdedb.dump」を、バックアップリストファイル名として「pg_dump_tdedb.list」を指定しています。

```
CMD> pg_restore -l pg_dump_tdedb.dump > pg_dump_tdedb.list
```

   b. 透過的暗号化機能で使用しているオペレータのリストファイルを作成するため SQL ファイルを作成します。

*4 pg_dump コマンドの詳細な利用方法は PostgreSQL マニュアルをご確認ください。

```
\t
\a
SELECT a.oid,
       'OPERATOR',
       c.nspname ,
       regexp_replace(oprname, '\*','\*'),
       rolname
  FROM pg_operator a
  JOIN pg_authid b
    ON a.oprowner=b.oid
  JOIN pg_namespace c
    ON c.oid=a.oprnamespace
 WHERE a.oprleft IN (SELECT oid
                       FROM pg_type
                      WHERE typname like 'encrypt_%')
    OR a.oprright IN (SELECT oid
                        FROM pg_type
                       WHERE typname LIKE 'encrypt_%');
```

c. 透過的暗号化機能で使用しているオペレータのリストファイルを作成します。この手順は透過的暗号化機能を利用するデータベースに psql で接続します。

次の例では、透過的暗号化機能を利用するデータベースとして「tdedb」を、前の手順で作成したオペレータのリストファイル作成用 SQL ファイルとして、「tde_operator.sql」を、オペレータリストファイル名として「tdedb_operator.list」を指定しています。

```
CMD> psql -d tdedb ^
More? -f tde_operator.sql -F " " ^
More? | findstr /R [0-9] > tdedb_operator.list
```

d. 作成したバックアップリストファイル（本節の手順では pg_dump_tdedb.list）とオペレータリストファイル（本節の手順では tdedb_operator.list）、そして Transparent Data Encryption for PostgreSQL Enterprise Edition インストール媒体に同梱されている透過的暗号化機能オブジェクトリストファイル (upgrade/installed_list_v1_3_0.txt)を使用して、ユーザデータリストファイルを作成します。

透過的暗号化機能オブジェクトリストファイルの命名規則は以下の通りですので、Transparent Data Encryption for PostgreSQL の旧バージョンとバージョンが合致する透過的暗号化機能オブジェクトリストファイルをご利用ください。

表 6-2 透過的暗号化機能オブジェクトリストファイルの命名規則

Edition	命名規則
Enterprise Edition	installed_list_vX_Y_Z.txt
Free Edition	installed_list_fe_VX_Y_Z_T.txt

X_Y_Z はアップグレード前の Transparent Data Encryption for PostgreSQL バージョンです。X.Y はメジャーバージョン、Z はマイナーバージョンを示します。T は Free Edition のみのリビジョン情報を示します。

次の例では、オペレータリストファイルとして前の手順で作成した「tdedb_operator.list」、バック アップリストファイルとして先ほど作成した「pg_dump_tdedb.list」、ユーザデータリストファイル名として「tdedb_user_data.list」を指定しています。

```
CMD> findstr /V /G:installed_list_v1_3_0.txt pg_dump_tdedb.list > tdedb_
user_data.tmp
CMD> findstr /V /G:tdedb_operator.list tdedb_user_data.tmp > tdedb_user_
data.list
```

注

複数の透過的暗号化機能を利用するデータベースがある場合はバック アップリストファイルとオペレータリストファイルの対応付けにご注意く ださい。

5. Transparent Data Encryption for PostgreSQL のアップグレードも同時に行っており、旧バージョンの Transparent Data Encryption for PostgreSQL が不要な場合、「第 5 章　アンインストール（16 ページ）」を参考にアンインストールします。アンインストールを行わない場合は、「5.2　透過的暗号化機能を対話型で無効化する方法（16 ページ）」のみ実施します。
6. ここまでの手順が完了後、透過的暗号化機能を利用するデータベースを削除します。

```
CMD> dropdb tdedb
```

7. PostgreSQL のメジャーバージョンアップグレードを行います。バージョンアップ手順については本節下部の関連リンクをご確認ください。
8. メジャーアップグレード後の PostgreSQL で透過的暗号化機能を利用するデータベースを作成します。

```
CMD> createdb tdedb
```

9. 「第 4 章　新規セットアップ（6 ページ）」を参考にメジャーアップグレード後の PostgreSQL に透過的暗号機能をインストールします。
10. 透過的暗号化機能を利用するデータベースに対してバックアップファイルと作成したユーザデータリストファイルを使用し、pg_restore コマンドでリストアします。

    次の例では、バックアップファイルに「pg_dump_tdedb.dump」を、リストア対象のデータベースとして「tdedb」を、作成したユーザデータリストファイルには「tdedb_user_data.list」を指定します。

```
CMD> set PGOPTIONS=-c encrypt.enable=off
CMD> pg_restore -d tdedb ^
More? -L tdedb_user_data.list -e pg_dump_tdedb.dump

CMD> set PGOPTIONS=
```

注

透過的暗号化機能を利用するデータベースが複数ある場合は、バックアップファイルとユーザデータリストファイルの対応付けにご注意ください。

11. リストアが成功しても、暗号鍵が登録されていないため、ユーザデータを復号して参照することはできません。そのため旧バージョンで利用していた最新の暗号鍵と同じ暗号鍵をリストアしたデータベースに登録する必要があります。また登録する暗号鍵は暗号化アルゴリズムも一致している必要がある点にご注意ください。

```
CMD> "C:\Program Files\NEC\TDEforPG96\bin\pgtde.bat" -m regist ^
More? -conf "C:\Program Files\NEC\TDEforPG96\conf\pgtde_secuser.properties"
Key management mode is not yet set.
Please select key management mode:
1. Simple TDE mode.
2. Standard TDE mode.
1
Enter new data key:
Retype new data key:
Select algorithm:
1. aes
2. bf
1
Are you sure you want to Regist new key to "tdedb"(DATABASE) with "aes" algori
thm? (Press Y(y) key to execute): Y
New key version 1 is registered to tdedb
```

関連リンク

PostgreSQL アップグレード手順

付録 A. セットアップ機能で出力されるエラーメッセージ

セットアップ機能で表示されるエラーメッセージについて説明します。

A.1 コマンドエラーメッセージ

セットアップ機能 cipher_setup.bat（cipher_setup.ps1）で表示されるエラーメッセージの一覧を下記に記載します。

表 A-1 Windows 版エラーメッセージ一覧

エラーメッセージ	対処方法
Internal error occurred	内部エラーが発生しています。システム管理者に連絡を行ってください。
File does not exist: <ファイル名>	インストールしたファイル構成が破損している可能性があります。Transparent Data Encryption for PostgreSQL の再インストールを実行してください。
Must be superuser to execute this action.	接続ユーザは PostgreSQL のスーパユーザを指定してください。
Could not connect to the database.	接続情報の内容を確認してください。
Could not use template1 database.	「template1」以外のデータベースを指定してください。
Security user must not be super user.	セキュリティ管理ユーザにはスーパユーザではないユーザを指定してください。
Security user could not access to database.	データベースに接続できませんでした。接続情報の内容を確認してください。
The length of Port must not be zero.	ポート番号には空文字以外を入力してください。
Port must be integer.	ポート番号には整数を入力してください。
The length of Database must not be zero.	データベース名には空文字以外を入力してください。
The length of Superuser must not be zero.	スーパーユーザ名には空文字以外を入力してください。
The length of Database Password must not be zero.	スーパーユーザのパスワードには空文字以外を入力してください。
The length of Security User must not be zero.	セキュリティ管理者ユーザ名には空文字以外を入力してください。
The length of Security User Password must not be zero.	セキュリティ管理者ユーザのパスワードには空文字以外を入力してください。
Transparent data encryption feature has not been activated yet.	透過的暗号化機能が有効化されているデータベースに対して再実行してください。
Lock file does not exist. File name: $env:INSTALLFILE	表示されたファイルをリストアするなど復旧の上、cipher_setup.bat を再実行してください。復旧方法についてはシステム管理者に連絡を行ってください。
Lock file already exists. File name: $env:INSTALLFILE	既に対象データベースは透過的暗号化機能が有効になっているため、有効化は不要です。
Transparent data encryption function has already been activated.	既に対象データベースは透過的暗号化機能が有効になっているため、有効化は不要です。
Could not activate transparent data encryption feature.	透過的暗号化機能の有効化に失敗しました。出力されたエラーメッセージファイルを確認してください。
Could not inactivate transparent data encryption feature.	透過的暗号化機能の無効化に失敗しました。出力されたエラーメッセージファイルを確認してください。

付録 B. ディレクトリ・ファイル構成

表 B-1 Windows ディレクトリ・ファイル構成

ディレクトリ・ファイル構成			説明
TDEforPG<XX>\ XX は PostgreSQL メジャーバージョン	bin\	cipher_setup.bat	透過的暗号化機能セットアップ起動バッチ
		cipher_setup.ps1	透過的暗号化機能セットアップスクリプト
		pgtde.bat	暗号化機能実行コマンド
	conf\		
	lib\	pgcrypto.dll	透過的暗号化機能用ライブラリ
		data_encryption.dll	透過的暗号化機能用ライブラリ
	lib\init		透過的暗号鍵機能内部実行スクリプト群
	lib\conf		透過的暗号化機能内部設定ファイル群
	lib\prop		透過的暗号化機能定義ファイル群
	lib\jar		透過的暗号化機能実行基盤ファイル群
	lib\psql	libpq.dll	PostgreSQL 接続用ライブラリ
		psql.exe	内部コマンド発行用 PostgreSQL クライアントプログラム
	jre\		透過的暗号化機能用 Java 実行環境
	template\	pgtde.properties.template	透過的暗号化機能コマンド pgtde 用設定ファイルのテンプレート
	log\		Transparent Data Encryption for PostgreSQL 用のデフォルトログ出力先
	sys\		透過的暗号化機能システム管理用ディレクトリ
	LICENSE		利用しているオープンソースライセンスについて

付録 C. 改訂履歴

本マニュアルの改訂履歴は以下のとおりです。

表 C-1 改訂履歴一覧

版数	発行日	改訂履歴
初版	2018 年 4 月	初版作成
第二版	2018 年 7 月	• 第 6 章 アップグレードの追加 • エラーメッセージの修正(付録 A. セットアップ機能で出力されるエラーメッセージ)
第三版	2020 年 9 月	• Windows Server 2019 に対応したことを追記(第 3 章 動作環境の確認とインストール前の準備) • PostgreSQL 11 に対応したことを追記(第 3 章 動作環境の確認とインストール前の準備)

マニュアルコメント用紙

読者各位

説明書に関するご意見、ご要望、内容不明確な部分について具体的にご記入のうえ、販売店または、当社担当営業、担当ＳＥにお渡しください。		お客様ご提出日	年　月　日
		〒 ご住所	〒
マニュアルコード	ＯＳＳＤＢＴＤＥ０６－０３	貴社名 所 属	
マニュアル名	Transparent Data Encryption for PostgreSQL セットアップカード（Windows 版）	お名前	

項番	ﾍﾟｰｼﾞ	行・図番	指摘区分	指 摘 内 容	添付資料
1					

備考　指摘区分　１：誤り　２：誤字・脱字　３：難解　９：ご要望

ご協力ありがとうございます。

（注意）販売店員または、当社営業部員、ＳＥは、すみやかに所定の手続きに従ってマニュアル担当までお送りください。（メール：２２－Ａ０７０３）

なお、ＮＥＣメールがない場合は、下記まで郵送してください。

〒２１１－８６６６ 神奈川県川崎市中原区下沼部１７５３

日本電気(株) AI プラットフォーム事業部 SDP グループ宛

販売店員 営業部員 ＳＥ記入	販売店名 または 所属名		担当		ﾒｰﾙ TEL	

ＮＥＣ

Transparent Data Encryption for PostgreSQL Enterprise Edition
セットアップカード
（Windows 版）

ＯＳＳＤＢＴＤＥ０６－０３

2020 年 09 月 第三版 発行

日本電気株式会社