Orchestrating a brighter world NEC

DBSC早春セミナー

OSS データベースへの暗号化の取り組み
～ PostgreSQLの透過的暗号化とサポートサービス ～

２０２２年3月

日本電気株式会社
白石雅己

目次

- PostgreSQLのセキュリティ機能について
- 何故、データベースの暗号化が必要か
- Transparent Data Encryption for PostgreSQL(TDEforPG)とは
- pgcryptoとの比較
- 暗号化機能の導入にあたって
- 暗号化状態の推移
- 安心してご利用頂くために
  - PostgreSQL保守サポートサービスについて
- 導入例
- まとめ

PostgreSQLのセキュリティ機能について

項目		内容	評価
認証		パスワード認証に加え、OS認証、LDAPやRADIUSなどの外部サーバと連携した認証、GSSAPI認証などをサポート	○
アクセス制御	接続元やユーザごと	・ユーザ、接続元、接続先DB、認証方法の組み合わせでアクセス制御 ・時間（時間帯、連続接続時間）により**制御する機能は無い**	△
	表・列・行レベル	ユーザに対する表・列へのアクセス権限の設定に加え、 行レベルのセキュリティポリシーの定義が可能	△
通信データの暗号化		SSLによる通信経路の暗号化	○
格納データの暗号化		pgcryptoモジュールの**関数を使用した暗号化**	△
監査ログ		・SQL文、ログイン成功/失敗などのログ出力が可能 ・コマンド種別ごとの実行ログや、データの更新履歴をログとして**記録する機能は無い** ・目的ごとのログ**出力先変更は不可**	△
特権ユーザからのデータ保護		ロールにより細かく権限の管理・制御が可能だが、スーパユーザであれば**権限チェックを行わない**	△

何故、データベースの暗号化が必要か

情報漏洩の代償

- ✓ 信用低下
- ✓ コストアップ
  謝罪、再発防止の対応
- ✓ 罰金や取引停止、刑事訴追
- ✓ 知的財産の情報が競合企業の手に渡り、競争で不利に

どのような情報が在るか

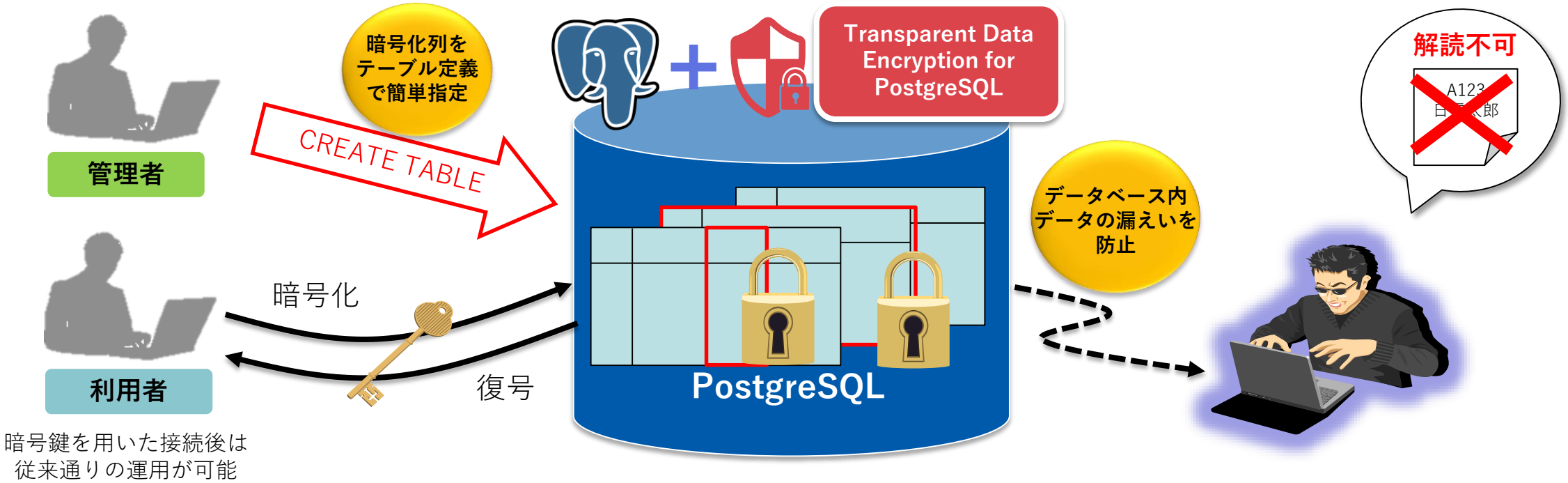
- ✓ 個人情報
  マイナンバー、氏名、住所・・・・
- ✓ 知的資産
- ✓ 財務情報
  カード番号、口座番号・・・
- ✓ 医療記録
- ✓ 購入履歴
- ✓ 会話(コールセンターログ)

法律・業界規制等

- ✓ 個人情報保護法
- ✓ 不正競争防止法
- ✓ マイナンバー法

Transparent Data Encryption for PostgreSQL(TDEforPG)とは

- PostgreSQLに透過的暗号化機能を付与する製品です
- 2015年にOSSとして公開し、NECによるサポートと付加機能をセットにした有償版を別途展開しています
- 暗号化方式としては列単位と行単位の二種類をご用意しています



暗号化方式

◆ 暗号化方式の違い

■**列単位暗号化**：暗号化したい列を指定して暗号化する方式

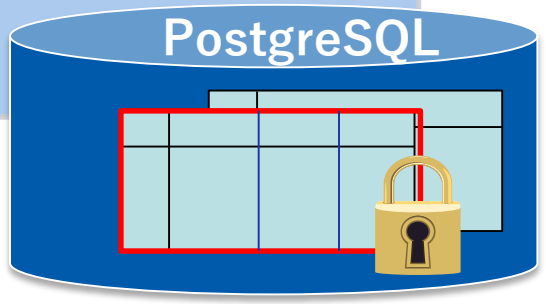
■**行単位暗号化**：テーブルに格納されるすべてのデータを暗号化する方式

列単位暗号化

- 暗号化したい列を指定
  （複数列も指定可）
- 暗号化対象列に対してB-tree
  インデックスが使用不可
  ※ハッシュインデックスのみ使用可
- PostgreSQL 9.5～12に対応

行単位暗号化

- テーブル全体の暗号化
- すべての列に対してB-tree
  インデックスが使用可
  ※インデックス部分は暗号化
  されない
- PostgreSQL 12のみ対応



2種類のEdition

◆ 無償のOSS版と商用版から選択が可能

- 基本的な機能が利用可能なFree Editionと、
  強化された暗号化機能に保守サポートを含むEnterprise Editionを展開しています
- Enterprise Editionはサブスクリプションライセンス（1年更新）での提供となります

Free Edition（OSS版）	• ＧＰＬライセンスで利用可能 • 2種類のデータ型を暗号化可能(TEXT、BYTEA) • GitHubで公開中 • 商用版へのアップグレードが可能
Enterprise Edition（商用版）の強化項目	• PostgreSQL本体も含めた保守サポートを提供 • 暗号化データ型を追加(NUMERIC,TIMESTAMP,INTEGER) • 複数バージョンの暗号鍵に対応 • AWS KMSを利用した鍵管理(列単位暗号化でLinuxのみ) • 行単位暗号化を提供 • アプリケーション改修が不要な簡易動作モードを選択可能

※ AWS はAmazon Web Services、KMSはKey Management Serviceの略称です。
アマゾン ウェブサービス、Amazon Web Services、Amazon EC2およびAmazon Web Servicesロゴは、Amazon.com, Inc.またはその関連会社の商標です。

Edition別の機能差異　比較一覧

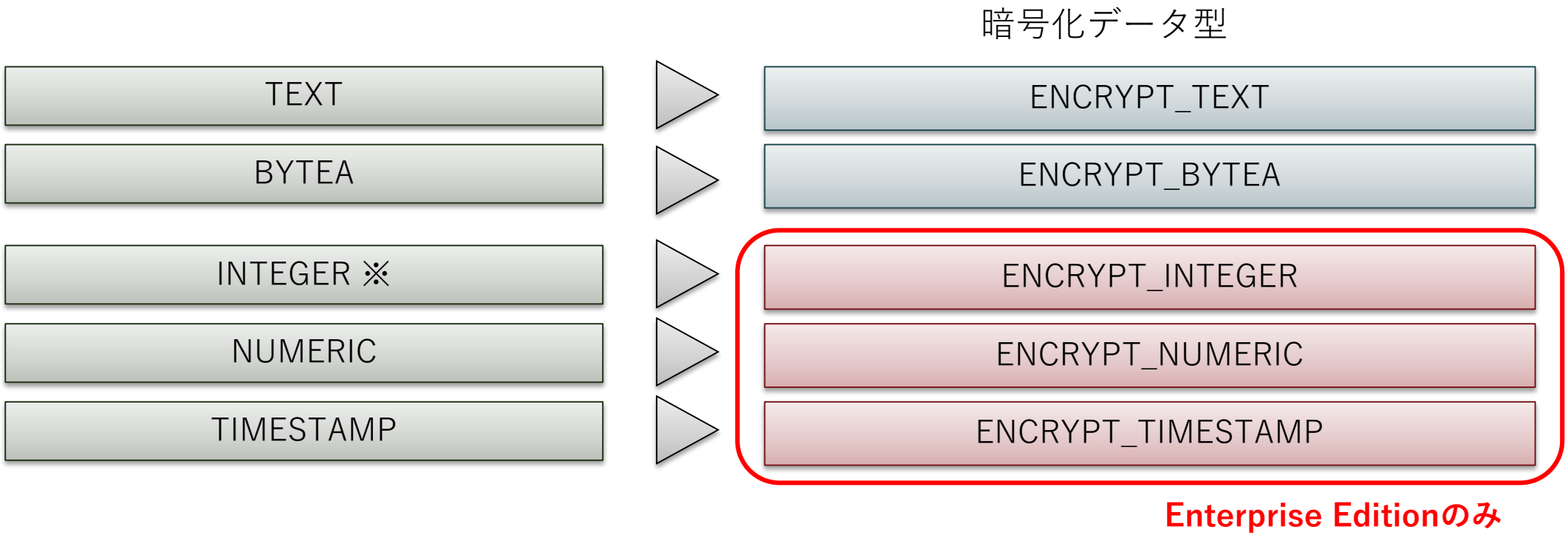
機能／サービス		Enterprise Edition		Free Edition
		列単位	行単位	
利用可能な暗号化データ型	テキスト	○	－	○
	バイト列（画像など）	○		○
	数値	○		×
	整数	○		×
	日付・時刻	○	行単位での暗号化のためデータ型での指定なし	×
鍵管理機能	鍵の変更に伴うバージョン管理機能	○	○	× 暗号鍵のバージョン管理機能なし
	Amazon Web Service 鍵管理機能（Key Management Service）とのシームレスな連携	○ (Linux版のみ)	×	×
	簡易動作モードの利用	○	○	×
サポート・サービス				
Transparent Data Encryption for PostgreSQLのPPサポートサービス		○	○	×
PostgreSQL本体のサポートサービス		○	○	×

利用可能な暗号化データ型

列単位

◆ 利用可能なデータ型が増加

■ 列単位暗号化では5種類のデータ型を暗号化データ型として提供しており、
うち3種類のデータ型はEnterprise Editionでのみ利用可能です



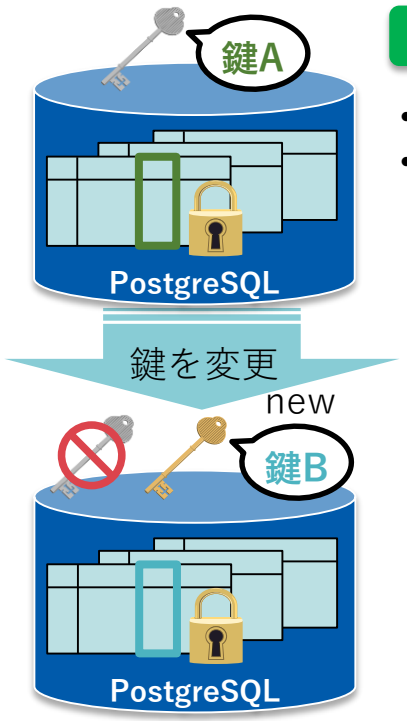
※ SMALLINT、INTEGER、BIGINT が対象となります。

鍵の変更に伴うバージョン管理機能

列単位　行単位

- 複数Verの鍵を保持し、鍵変更時の待ち時間をなくすことが可能に
  - Free Editionでは鍵を１つしか持てず、鍵を変更すると新しい鍵による全データの再暗号化が必要であり、完了するまで排他ロックがかかります
  - Enterprise Editionは複数の鍵を内部で保持することが可能であり、全データの即時再暗号化を必要としません。運用を止めずに鍵を変更できます

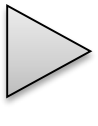
列単位暗号化の構成例



Free Edition

- 鍵の変更とともに鍵Aを破棄
- 鍵Bによる**全データの再暗号化完了まで利用不可**

データ量が多い程、待たされる



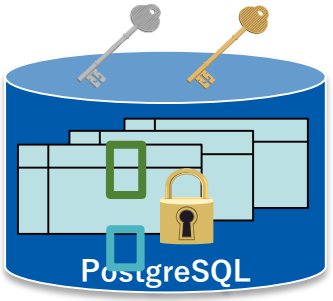
	氏名	所属
…	…	…
00128	○△×？	営業部
…	…	…
00653	■A%X	技術部

new

鍵を変更しても待たされない

Enterprise Edition

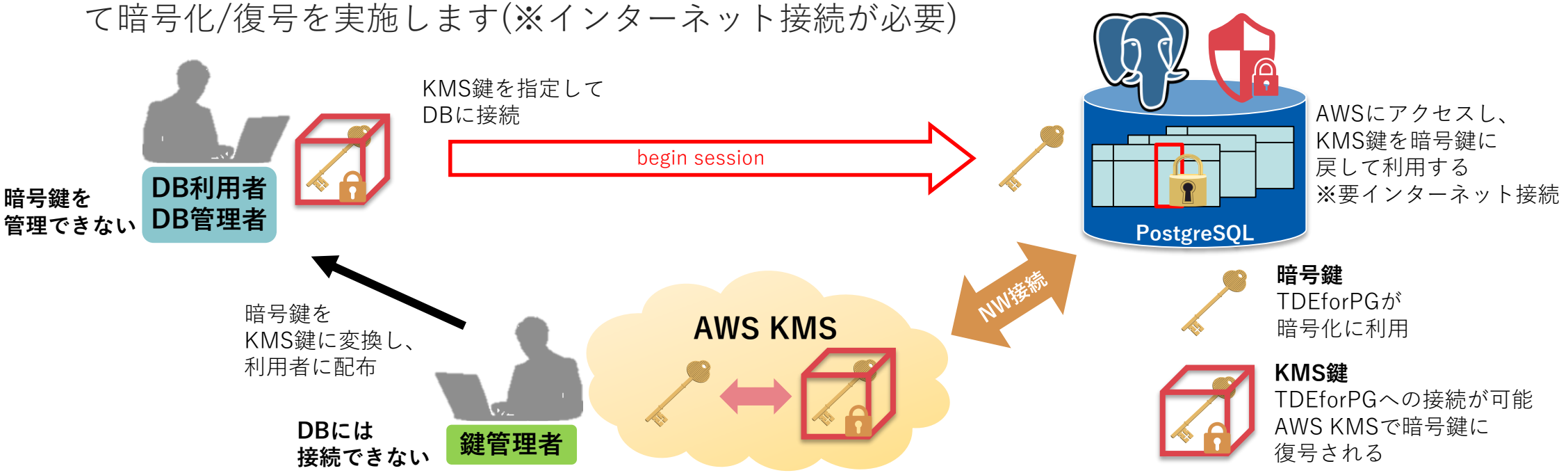
- 各Verの鍵で暗号化されたデータが混在
- **最新の鍵のみ接続可**、全データの復号可
- レコードの更新時に最新の鍵で再暗号化



Amazon Web Service 鍵管理機能（Key Management Service）とのシームレスな連携

列単位

- より安全な鍵管理を実現できる、AWS KMSと連携可能
  - 対象は列単位暗号でLinux環境のみ
  - AWS提供の鍵管理機能 AWS KMSと連携し、
    鍵管理者とDB管理者で権限を分離させる運用が可能です
  - 鍵管理者はAWS KMSを用いてTDEforPGの暗号鍵をKMS鍵に変換します
  - DB利用者/管理者はKMS鍵を用いてDBに接続すると、DB側はAWSにアクセスし、KMS鍵を暗号鍵に戻して暗号化/復号を実施します(※インターネット接続が必要)



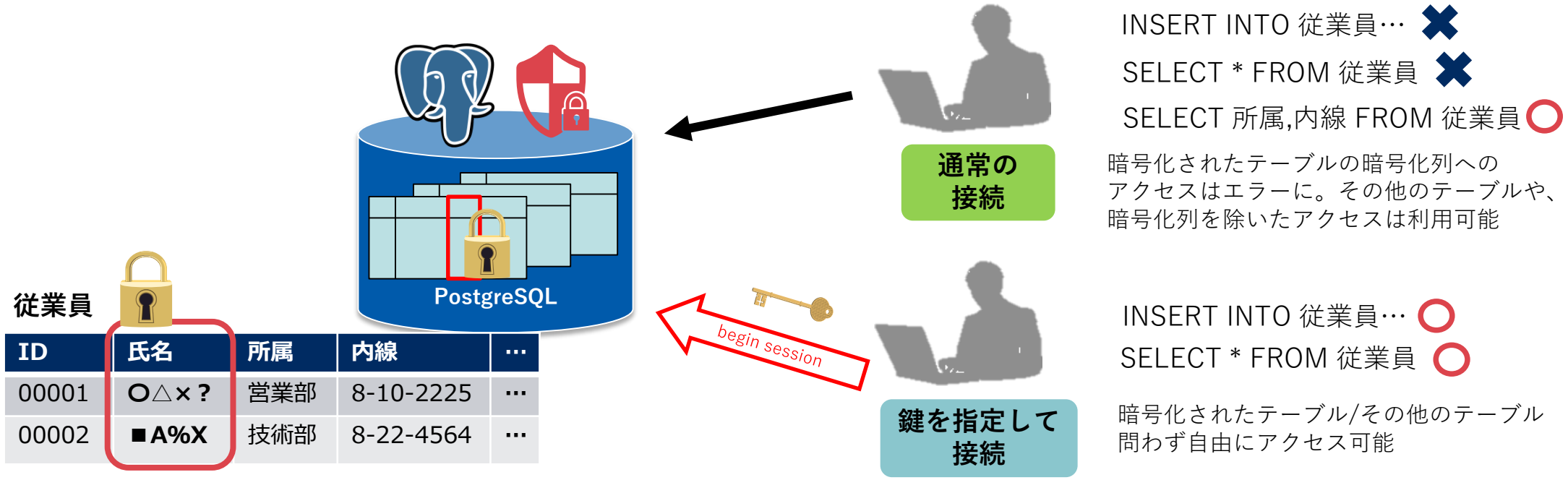
pgcryptoとの比較

比較項目	pgcrypto	TDEforPG	
		列単位	行単位
暗号化の単位	列	列	行
アプリケーションの変更	SQLが**対象列を操作する箇所で関数実行するよう変更が必要**	最初に暗号鍵の設定を行えば、**SQLを変更せず透過的に暗号化・復号処理が可能**	
暗号キー管理	アプリケーション側で管理	AWS KMS連携機能	アプリケーション側で管理
インデックス制約	**大小比較・キー順処理不可**	**大小比較・キー順処理不可** ハッシュインデックスのみ定義可(一本釣り化)	**大小比較・キー順処理可能** インデックスの暗号化不可
処理オーバヘッド	暗号化列数に依存	暗号化列数に依存	行全体を暗号化するため範囲は広くなるが、処理回数は抑えられる

暗号化機能の導入にあたって

- CREATE TABLEまたはALTER TABLE(※1)を用いて暗号化対象を定義します
- 暗号化されたテーブルにアクセスする際は、鍵を指定した専用のセッションによる接続が必要となります
  ※通常の方法で暗号化列にアクセスした場合、エラーとなります
- 既存のAPはセッションに関する数行の改修が必要(※2)となります

※1 列単位の指定はALTER TABLEでも可能
※2 Enterprise Editionで簡易動作モードを利用する場合は不要



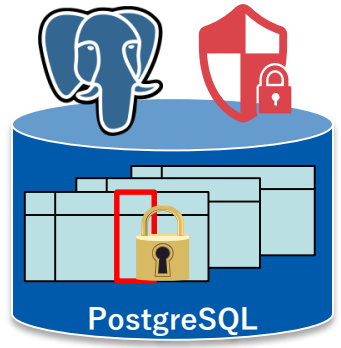
暗号化機能の利用イメージ（列単位）

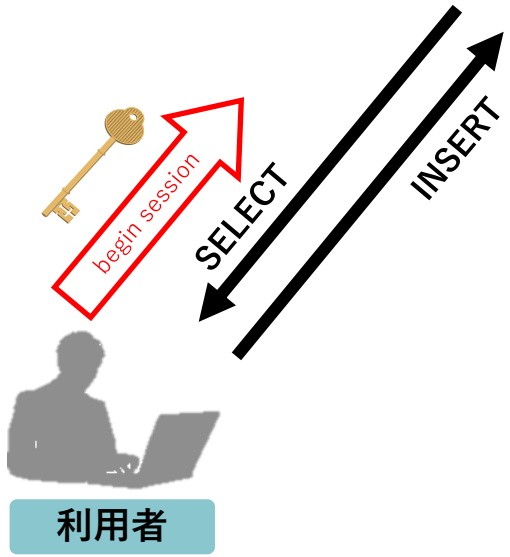
① 管理者があらかじめ、暗号化対象とする列を指定しておく

従業員

ID	氏名	所属	内線	メール
…	…	…	…	…
00128	〇△×？	営業部	8-10-2225	Vhdowy(¥a
…	…	…	…	…
00653	■A%X	技術部	8-22-4564	.,cu73xcv

暗号化列属性を
EXTENTIONとして実装





SELECT * FROM 従業員 WHERE id = 00128;

参照時は自動で復号

(00128,**菊地 新太**,営業部,8-10-2225,**s-kikuchi@mail.com**)

INSERT INTO 従業員 VALUES
(00653,田中 一,技術部,8-22-4564,h-tanaka@mail.com)

挿入/更新時に自動で暗号化

(00653, **■A%X**,技術部,8-22-4564,**.,cu73xcv**)

② 利用者は鍵を指定して接続した後、暗号化/復号を意識せずに通常のSQLでそのまま利用可能

暗号化機能(列単位)の利用イメージ

```
-- CREATE TABLE
CREATE TABLE Employee(
 EmployeeID Integer PRIMARY KEY,
 Name TEXT,
 Address ENCRYPT_TEXT,
 TelephoneNumber ENCRYPT_TEXT
);

-- START SESSION
SELECT cipher_key_disable_log();
SELECT pgtde_begin_session('cipherkey');
SELECT cipher_key_enable_log();

-- INSERT DATA
INSERT INTO Employee VALUES(1,'従業員1','滋賀','003-0001-0001');

-- SELECT ALL
SELECT * FROM Employee ;

-- END SESSION
SELECT pgtde_end_session();
```

暗号化列属性を指定したテーブル定義

鍵を指定した専用のセッションを開始
（ログ出力を一時的に無効化、暗号鍵は文字列で指定）

データの挿入（通常のSQL文）

データの検索（通常のSQL文）

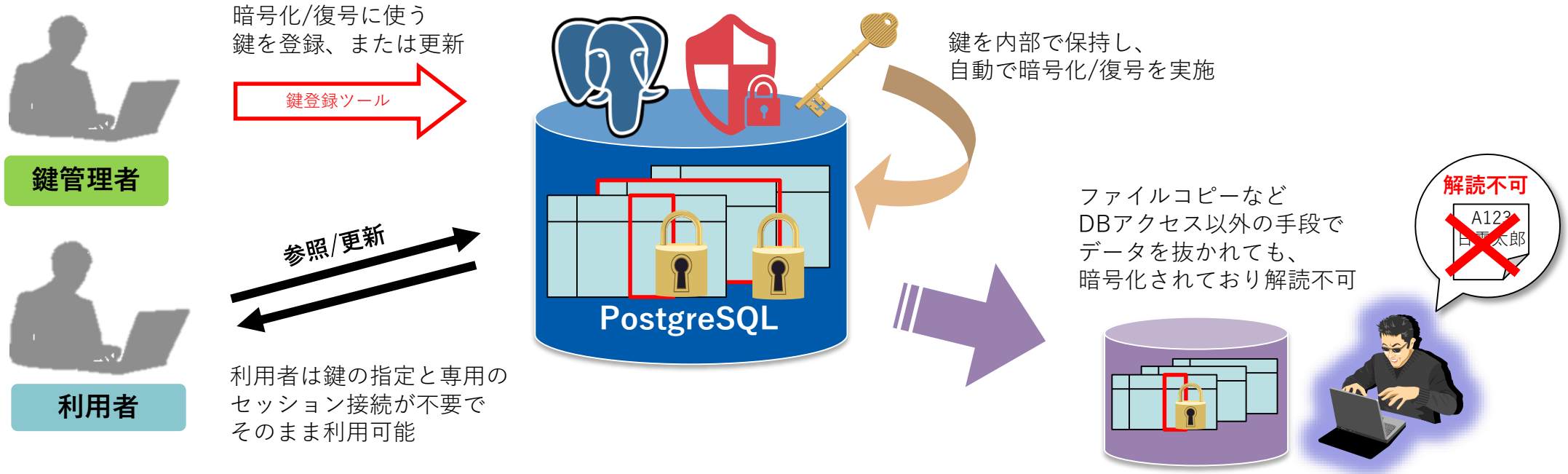
専用のセッションを終了

簡易動作モード

列単位　行単位

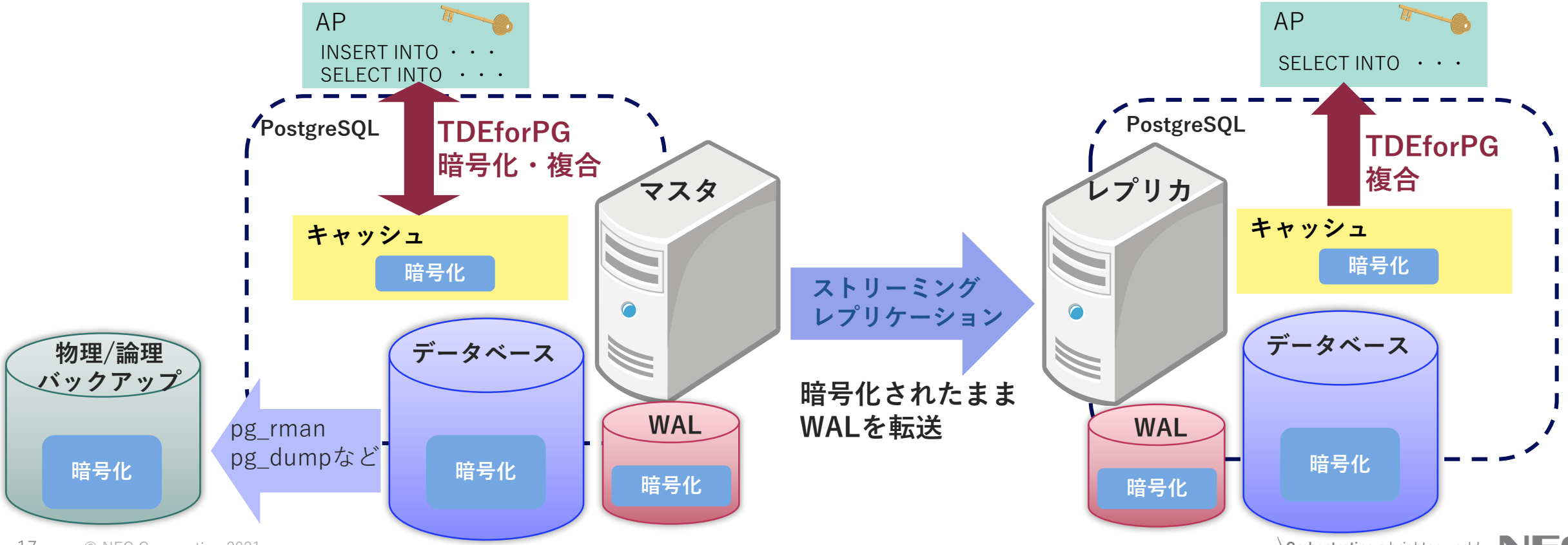
◆ アプリケーション改修なしで利用可能な簡易動作モードを搭載

- 簡易動作モードではDB内部に鍵を保持し、DBに接続したユーザに対し自動で暗号化/復号を実施するため、利用者は鍵の指定が不要となります
- その他の鍵管理機能より相対的にセキュリティレベルは低下しますが(DBアクセス可能なユーザはそのまま復号、参照が可能に)、アプリケーションの改修なしで利用可能です



暗号化状態の推移

- キャッシュ,データベース,WAL,バックアップ内は、全て暗号化されている
- ストリーミングレプリケーションは、暗号化されたWALを転送して
  レプリカ側データベースに反映
- 最初に暗号鍵情報をレプリカサイトに手動で持ち込む



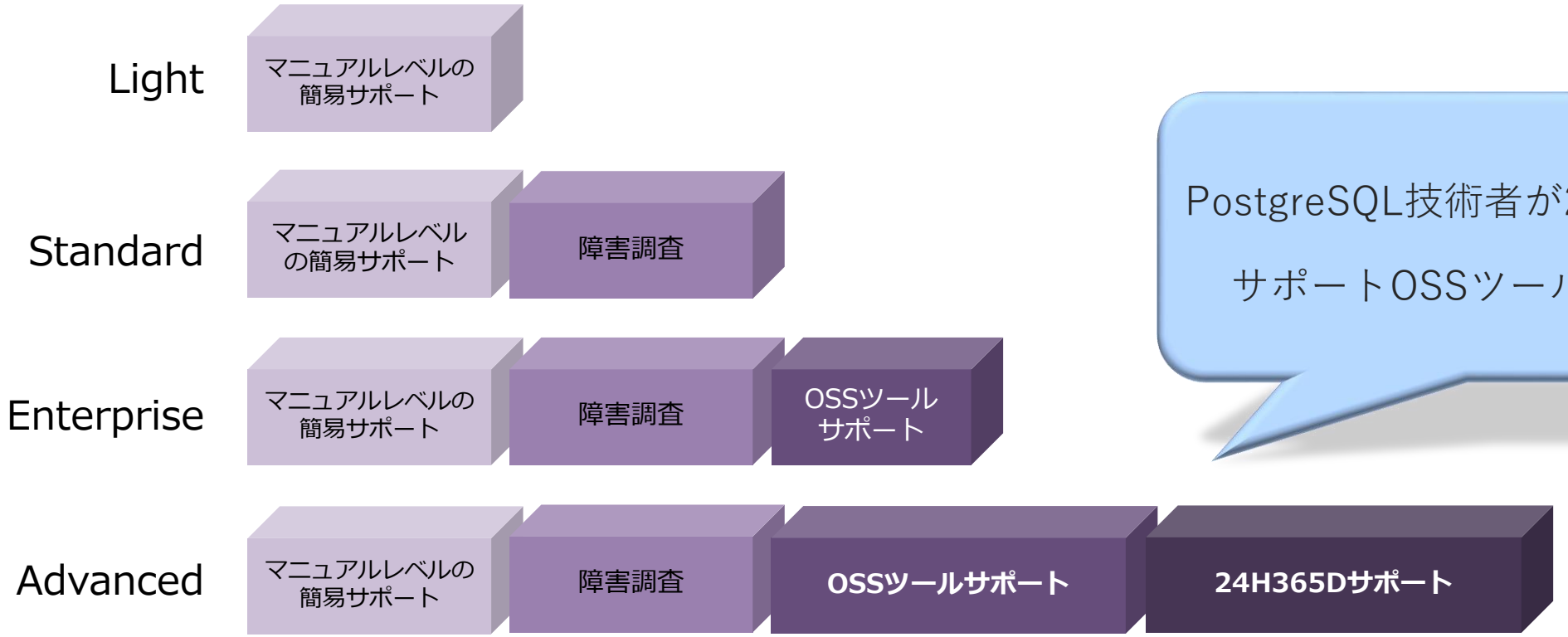
安心してご利用頂くために

TDEforPG Enterprise Editionの万全な保守のために、ご一緒にPostgreSQLの保守サポートに加入いただくことをお勧めしています

PostgreSQL保守サポートサービスについて

- PostgreSQLの技術的な問い合わせや障害発生時の調査などに対応
- 定額メニューで保守サービスを提供

◆ サポートサービスメニュー



PostgreSQL技術者が24時間365日対応

サポートOSSツールも大幅に拡充

保守サポートサービス比較のまとめ

		Standard	Enterprise	Advanced	サービスの内容
PostgreSQL QA対応		○	○	○	マニュアルレベルのPostgreSQL運用に関するQA対応
パッチの問い合わせ		○	○	○	コミュニティがリリースしたパッチに関するQA対応
運用ツールのQA対応		×	△ ※1	○ ※2	PostgreSQLの機能拡張ツール、運用ツールのQA対応 ※1 Enterpriseのみ重要性の高い機能拡張ツールのQA対応 ※2 pgpool-IIほかサポート対象OSSを追加
24H 技術者サポート		×	×	○	技術者による24H365Dのサポート提供
障害調査	メッセージの調査	○	○	○	PostgreSQLサーバが出力するメッセージに該当する既知問題の調査 ログメッセージの内容をさらに解析する場合はStandard以上のサービスが必要
	coreダンプの調査	○	○	○	PostgreSQLが出力したcoreファイルの調査
	再現環境の構築・評価	○	○	○	NECにて再現環境構築、評価
	コミュニティへのフィードバック	○ ※3	○ ※3	○ ※3	新規障害判明時、コミュニティに対する障害報告と措置への働きかけ ※3 コミュニティによる障害解決を保証するものではありません
パッチ作成		×	×	×	NEC独自のパッチ作成、提供
データの保障・復旧作業		×	×	×	ユーザデータの保障、復旧作業
パフォーマンス分析・チューニング		× (△)	× (△)	× (△)	PostgreSQLサーバの性能情報収集、分析、チューニング作業 ※プロフェッショナルサービスでの対応

PostgreSQL保守サポートサービスの対象範囲

PostgreSQL保守サポートサービス対象コンポーネント

全サービス共通

PostgreSQL標準機能

- 標準ツール（psql,pg_dump 等）
- contribパッケージ（pgcrypto,dblink等）
- **postgresサーバ**
- libpqライブラリ
- ecpgプリコンパイラ

外部API(※1)

- JDBCドライバ
- ODBCドライバ（psqlODBC）
- .NET Frameworkデータプロバイダ（Npgsql）
- PHP PostgreSQL拡張モジュール

独自開発プロジェクトの関連ツール(※2)

Enterprise対象

pg_hint_plan, pg_dbms_stats,pg_statsinfo, pg_bigm

Advanced対象

pg_hint_plan, pg_dbms_stats,pg_statsinfo, pg_bigm,pg_Admin,pgBadger,pgFincore,
pg_reporter,pg_rman,pg_bulkload,pg_repack,pgpool-II,oracle_fdw, orafce

（※１） 記載のないAPIドライバについてサポートをご希望の方はご相談ください
（※２） 対象外コンポーネントについて別途サポートをご希望の方は、ご相談ください

導入例：簡易動作モード利用によるAP改修最小化

背景：全社レベルで機密情報の暗号化が義務づけられPostgreSQLのデータも対象に

課題：既存APも多く、暗号化システム導入によるAPの影響を最小限に抑えたい

対応策

TDEforPGの**簡易動作モード**を利用することで
DB内部に鍵を保持し、DBに接続したユーザに対し
自動で暗号化/復号を実施するため、
利用者は鍵の指定が不要(AP改修不要)

※DBアクセス可能なユーザであれば誰でも、
参照が可能になるため、セキュリティレベルは低下

鍵を指定した専用のセッションを開始
（ログ出力を一時的に無効化、暗号鍵は文字列で指定）

```
-- START SESSION
SELECT cipher_key_disable_log();
SELECT pgtde_begin_session('cipherkey');
SELECT cipher_key_enable_log();

-- INSERT DATA
INSERT INTO Employee
    VALUES(1,'従業員1','滋賀','003-0001-0001');

-- SELECT ALL
SELECT * FROM Employee ;

-- END SESSION
SELECT pgtde_end_session();
```

専用のセッションを終了

まとめ

項目	内容
外部犯行による 不正アクセス防止	・ユーザ認証、ロールによるアクセス制御 ・表および列へのアクセス権限の設定 ・行レベルセキュリティポリシーの定義
内部犯行による 不正アクセス防止	・スーパーユーザ権限を持つユーザに対して内部犯行を防ぐ**仕組みが無い**
DB外で起こる盗聴・ 盗難防止	・通信暗号化(SSL対応) ・透過型データ暗号化 **（標準では関数を使用した暗号化のみ）** **・バックアップの暗号化機能はない**
事後の備え/不正アクセスの抑止力	・SQL文、ログイン成功/失敗に関しては、ログ出力可能 ・コマンドごとの実行ログや、データの更新履歴をログとして記録する機能無し

TDEforPG 導入で解決

- **暗号鍵がなければデータを複合して取り出せなくなった**
- **関数を使用しなくても透過的暗号化を実現**
- **暗号化された状態でバックアップ**

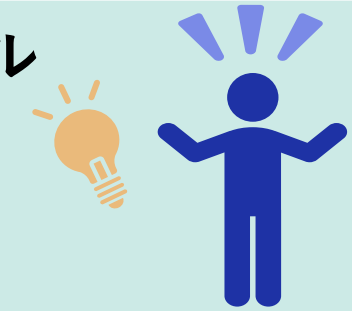
NECのPostgreSQLビジネス

2003年からサポートサービスを展開、数多くのPJを支援しており、
対応製品もご用意があります。

PostgreSQLサポートサービス

プロフェッショナルサービス

導入移行支援



本日紹介

保守サポート

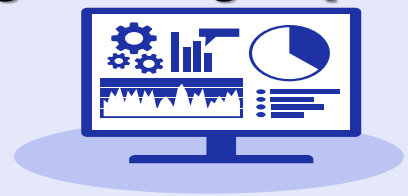
運用保守

本日紹介

TDE for PostgreSQL

透過的暗号化

WebSAM System Manager G PostgreSQL Monitoring Option



稼働監視

製品Webページ

◆ Transparent Data Encryption for PostgreSQL Enterprise Edition 製品Webページ
http://jpn.nec.com/tdeforpg/

■ Free EditionはGithubにて公開しております
https://github.com/nec-postgres/tdeforpg/

- Free Editionのライセンス（GPLで公開）
  https://github.com/nec-postgres/tdeforpg/blob/master/LICENSE

◆ PostgreSQLサポートサービス
https://jpn.nec.com/oss/middle_support/postgresql/

ご清聴ありがとうございました

\Orchestrating a brighter world

NECは、安全・安心・公平・効率という社会価値を創造し、
誰もが人間性を十分に発揮できる持続可能な社会の実現を目指します。


Orchestrating a brighter world
NEC