

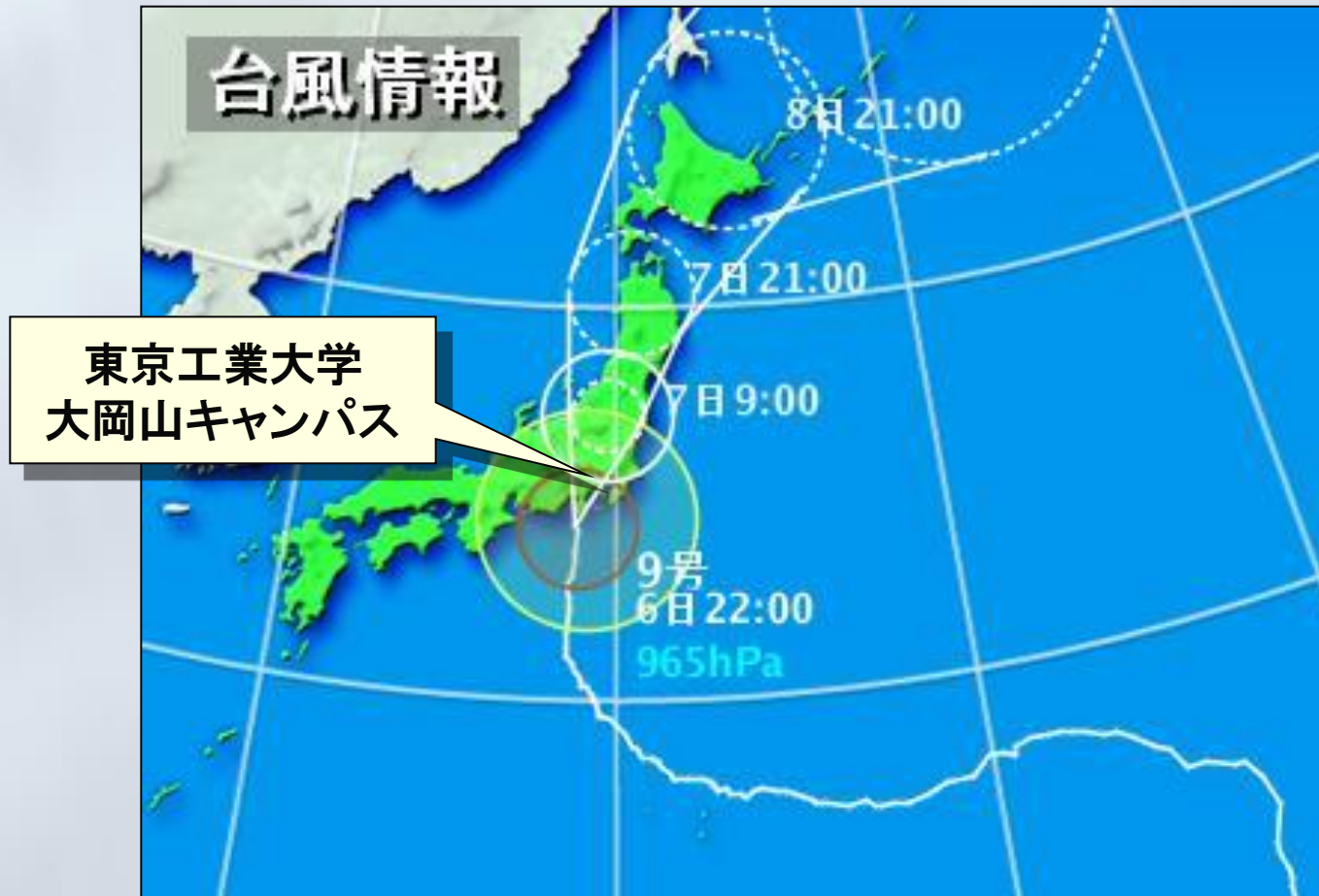
未踏ソフトウェア創造事業(2006下期)
最終成果報告会

The Security-Enhanced PostgreSQL

KaiGai Kohei <kaigai@kaigai.gr.jp>

はじめに

- 悪天候の中、ご参加ありがとうございます



本日のアジェンダ

- I. プロジェクトの概要・背景・思想
- II. SE-PostgreSQLアクセス制御方式
- III. デモンストレーション
- IV. SE-PostgreSQLの展開と未来

どんなプロジェクト？

■ SELinuxとの連携によって

PostgreSQLのアクセス制御機能を強化

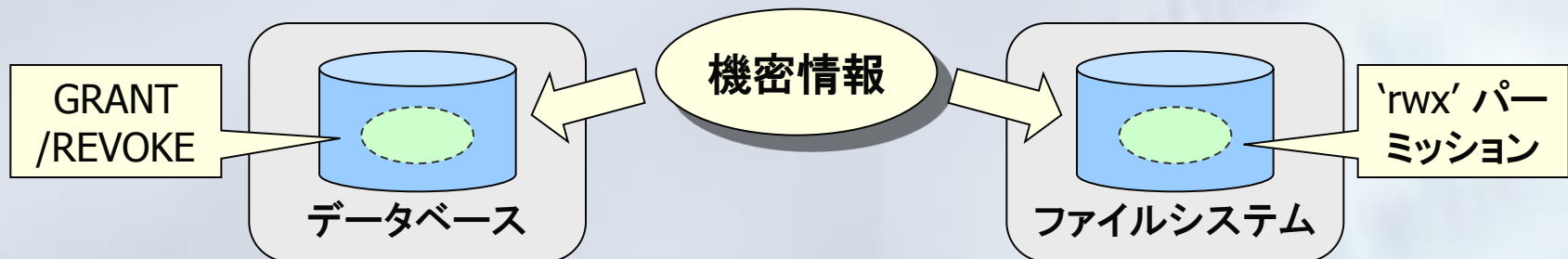
- ✓ データベースに対する、細粒度・強制アクセス制御
- ✓ OSと一体化した情報フロー制御
- ➡ 究極目標:『安心して利用可能なITインフラの提供』

■ プロジェクトの成果

- SE-PostgreSQL 8.2.4-1.0 正式版リリース ('07/09/03)
 - ✓ Linuxカーネルへの SE-PostgreSQL 対応機能の追加
 - ✓ コミュニティ標準ポリシーへの SE-PostgreSQL 対応の追加
- "The SE-PostgreSQL Security Guide" のリリース
- Fedora Projectへのマージ

“情報資産”を守る (1)

- 何を守りたいのか？
 - 情報システム
 - 情報資産
- 情報資産の格納手段とアクセス制御
 - ✓ “情報”を格納するには媒体が必要
 - ファイル ... 'rwx'によるパーミッション制御
 - データベース ... GRANT/REVOKEによるACL



“情報資産”を守る (2)

- SE-PostgreSQLの考え方
 - 情報の格納方法に拘らず、同じ情報には、同じアクセス制御が実施されるべき
 - ファイル、データベースの差異は『手段の違い』
- 共通のアクセス制御のために
 - 『共通の評価尺度』と『共通の意思決定システム』
 - “情報”のセキュリティ属性を抽象化/共通化
 - ✓ ラベル、セキュリティコンテキスト、etc
 - 一元化されたセキュリティポリシー
 - ✓ Security ServerとしてのSELinux

SELinuxとの連携 (1)

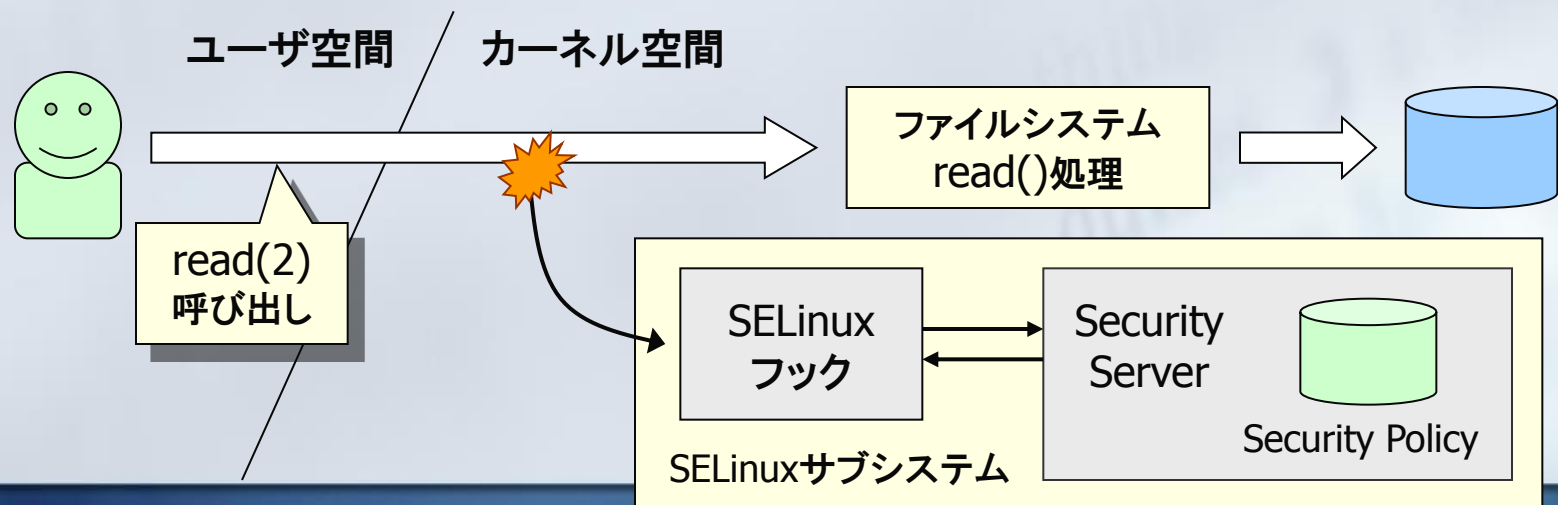
■ SELinux

■ OSのリファレンスモニタ

- System Call実行に対する強制アクセス制御

■ Security Server

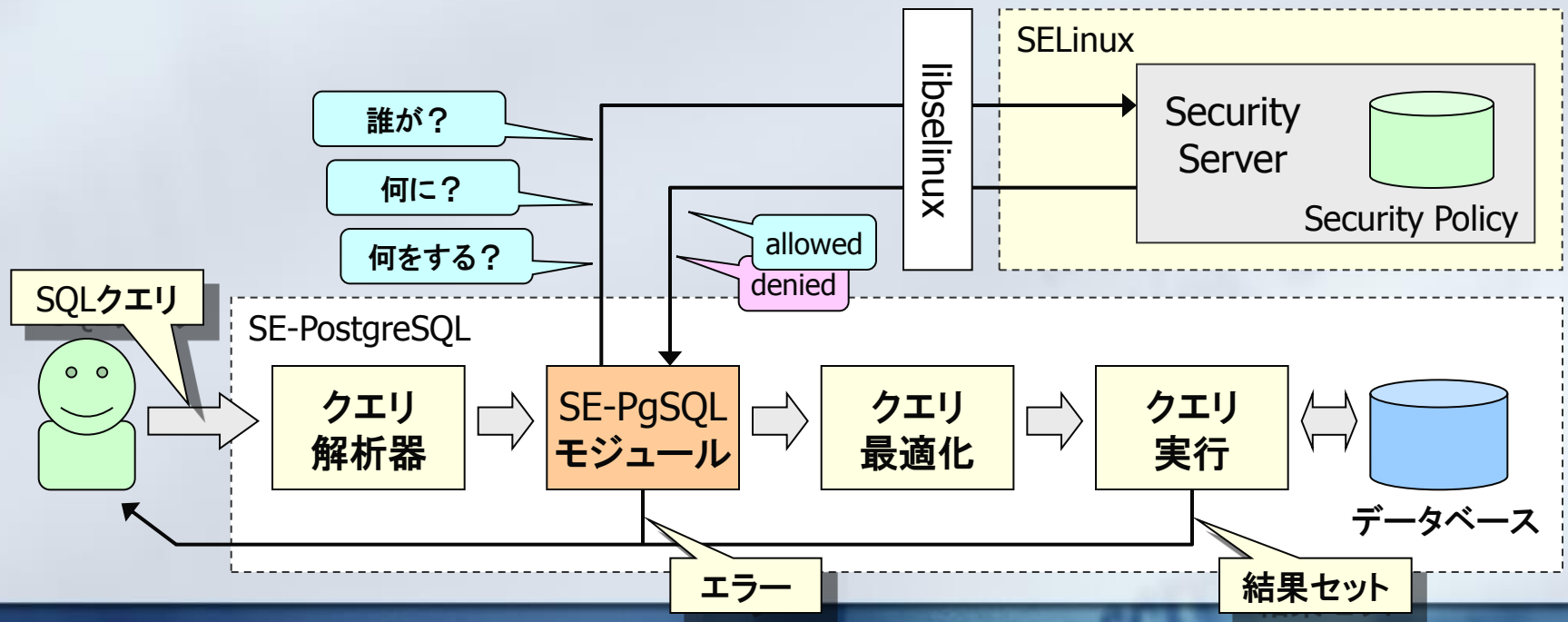
- セキュリティポリシーの管理
- 要求されたアクションの実行可否を回答



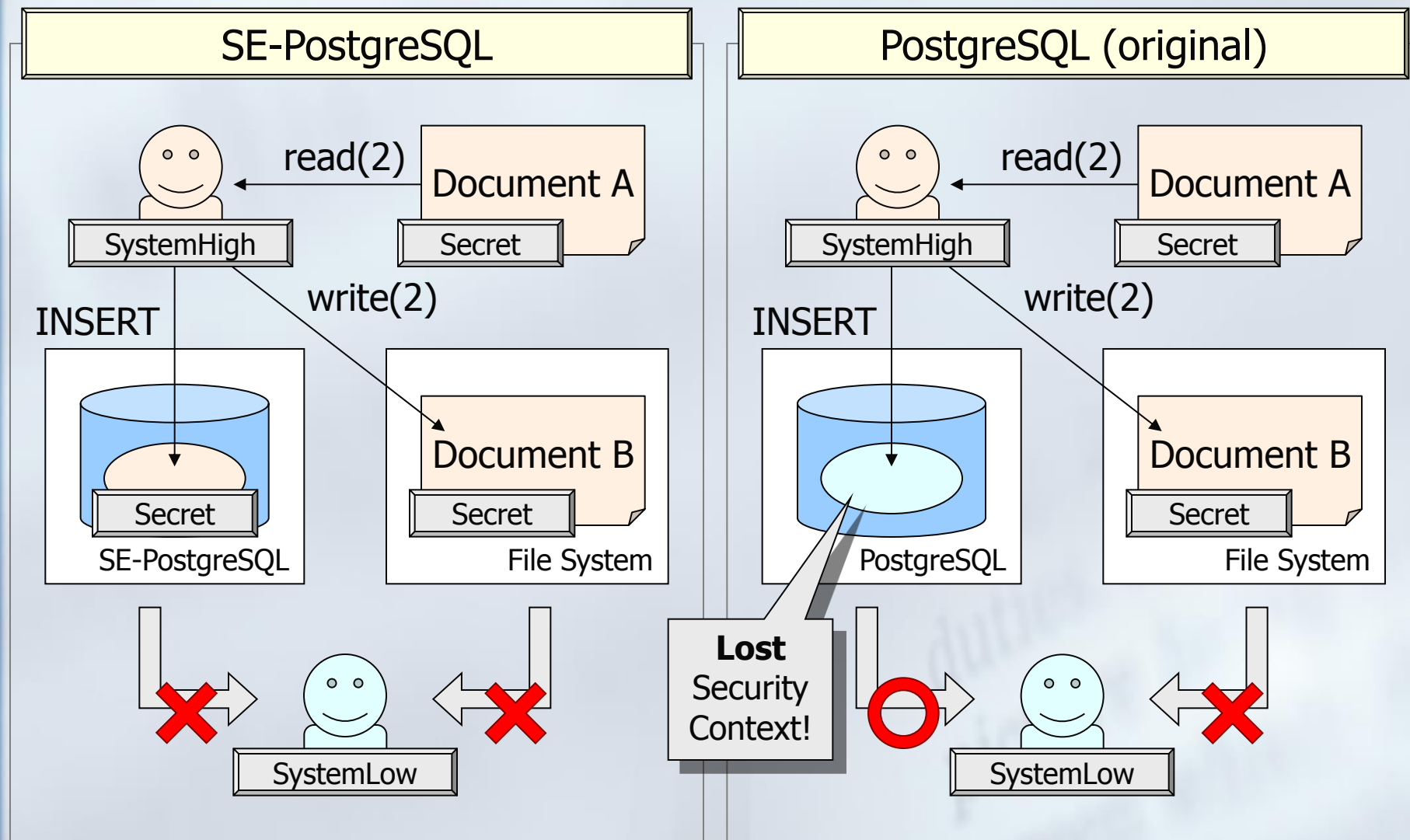
SELinuxとの連携 (2)

■ SE-PostgreSQL

- DBオブジェクトのセキュリティコンテキストを管理
- Security Serverに問い合わせ
 - ▶ 結果を利用してアクセス制御



情報フロー制御



本日のアジェンダ

- I. プロジェクトの概要・背景・思想
- II. SE-PostgreSQLアクセス制御方式
- III. デモンストレーション
- IV. SE-PostgreSQLの展開と未来

SE-PostgreSQLのアクセス制御

SQLクエリ

```
SELECT id, name, price FROM drink  
WHERE alcohol=true;
```



SE-PostgreSQL



drinkテーブル

| id | name | price | alcohol |
|----|---------|-------|---------|
| 1 | 'coke' | 110 | false |
| 2 | 'tea' | 120 | false |
| 3 | 'juice' | 150 | false |
| 4 | 'water' | 120 | false |
| 5 | 'wine' | 340 | true |
| 6 | 'beer' | 240 | true |

SQLクエリの検査

- ✓ 権限のないテーブル/カラムetcのアクセスを即エラーに

SQLクエリの書換え

- ✓ WHERE句に条件を追加、権限のないタプルを結果セットから除去

drinkテーブル

| id | name | price | alcohol |
|----|---------|-------|---------|
| 1 | 'coke' | 110 | false |
| 2 | 'tea' | 120 | false |
| 3 | 'juice' | 150 | false |
| 4 | 'water' | 120 | false |
| 5 | 'wine' | 340 | true |
| 6 | 'beer' | 240 | true |

SQLクエリの検査

- SQLクエリには...？
 - 対象のDBオブジェクトと、アクセス方法が記述されている
- SQLクエリの検査方法
 - クエリからDBオブジェクトを抽出、パーミッションを検査
 - 権限が不足していれば、即座にトランザクションをアボート
- 6種類のオブジェクトクラス、58種類のパーミッション

| db_database | db_table | db_column | db_tuple | db_procedure | db_blob |
|---|---|---|---|--|--|
| create drop getattr setattr relabelfrom relabelto access install_module load_module get_param set_param | create drop getattr setattr relabelfrom relabelto use select update insert delete lock | create drop getattr setattr relabelfrom relabelto use | relabelfrom relabelto use select update insert | create drop getattr setattr relabelfrom relabelto | create drop getattr setattr relabelfrom relabelto |

"標準セキュリティポリシー"に
既にマージされている

Case Study (1)

```
SELECT id, name, price * 1.05 FROM drink WHERE id in (3,4);
```

- idカラム ... db_column:{select use} 権限
- nameカラム ... db_column:{select} 権限
- priceカラム ... db_column:{select} 権限
- drinkテーブル ... db_table:{select use} 権限
- numeric関数 ... db_procedure:{execute} 権限
- numeric_mul関数 ... db_procedure:{execute} 権限
- 各タプルに対して ... db_tuple:{select use} 権限

SQLクエリの書き換え

- 行レベルアクセス制御の課題
 - 実際にSQLクエリを実行するまで、どのタプルにアクセスするのか予想できない
- 解決策
 - SQLクエリ実行時に、タプルへのアクセス権を評価
 - SQLクエリのWHERE句を書き換えて条件を追加
- 例
 - `SELECT * FROM drink WHERE alcohol = true;`
 - `SELECT * FROM drink WHERE alcohol = true
AND sepgsql_tuple_perms(...);`

各タプルのアクセス権を
評価するSQL関数

'security_context' システム列

■ 行レベルアクセス制御の課題

- 個々のタプルのセキュリティ属性をどのように表現するか？

■ 解決策

- 'security_context' システム列でセキュリティ属性を表現

- システム列:

- ✓ 全てのテーブルで自動的に作成される隠しカラム。
- ✓ 本来は Read-Only な属性
- ➡ 'security_context' は書込み可能なように拡張

■ 例

- `SELECT security_context, * FROM drink;`
- `INSERT INTO drink (security_context, id, name, price)`
`VALUES('system_u:object_r:sepgsql_ro_t', 7, 'milk', 130);`

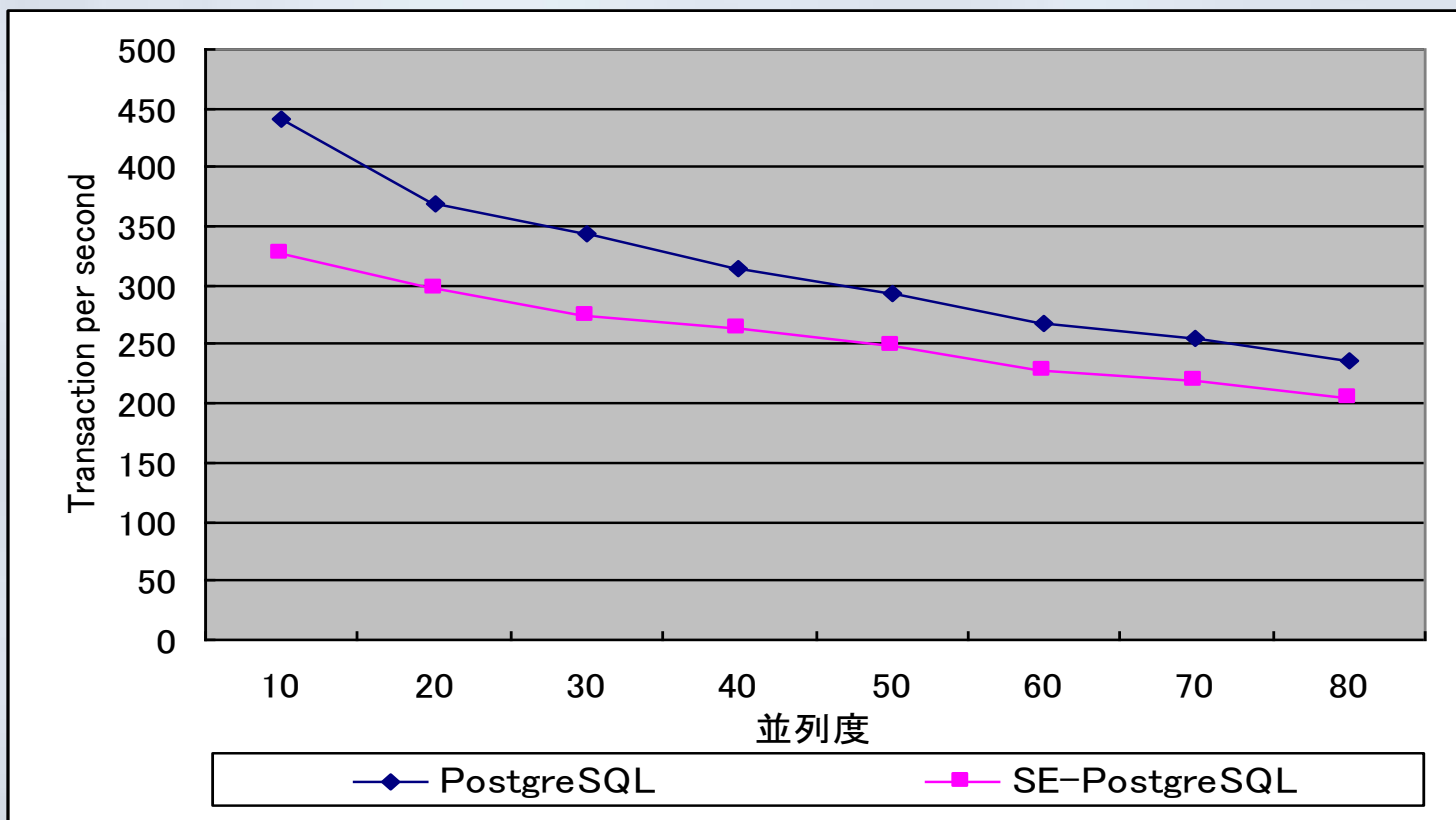
Case Study (2)

```
UPDATE drink SET name='sake', price = 2 * price WHERE id = 3;
```

- nameカラム ... db_column:{update} 権限
- priceカラム ... db_column:{select update} 権限
- idカラム ... db_column:{use} 権限
- drinkテーブル ... db_table:{select use update} 権限
- int4mul関数 ... db_procedure:{execute} 権限
- int4eq関数 ... db_procedure:{execute} 権限
- 各タプル ... db_tuple:{select use execute} 権限

パフォーマンス (ご参考)

- pgbenchによる測定
- Core2DuoE6400, Memory: 1GB, HDD: Serial-ATA
- Scaling Factor = 10, Number of total transaction = 120,000



※ shared_buffers = 512MB で測定、残りのパラメータはデフォルト値

Case Study (3)

```
CREATE TABLE t1 ( x integer primary key, y text);
```

- t1テーブル ... db_table:{create} 権限
- xカラム、yカラム ... db_column:{create} 権限

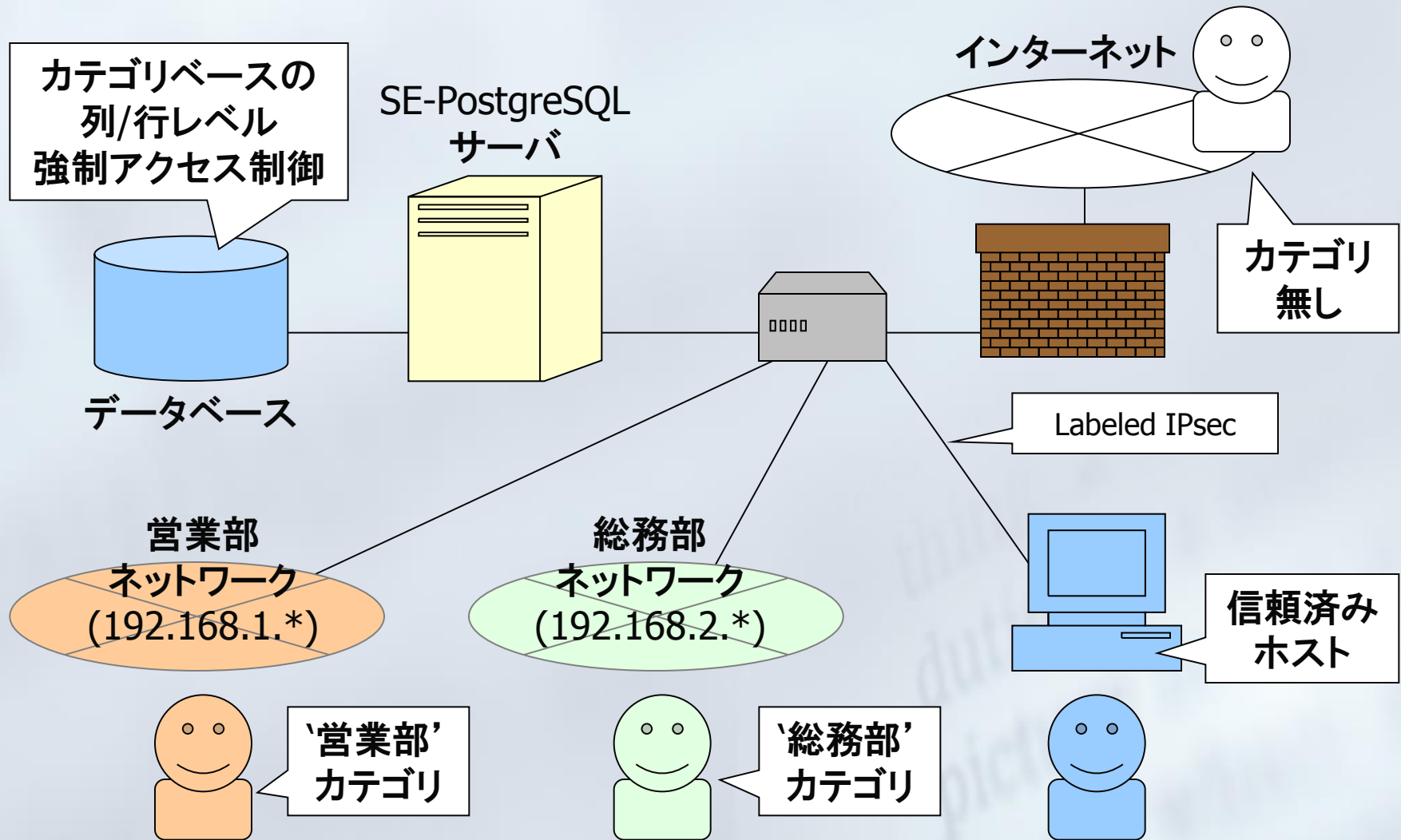
セキュリティポリシーは、新しいDBオブジェクトのセキュリティコンテキストを決定するルールを持っている。

- 全てのシステム列 ... db_column:{create} 権限
- pg_typeシステムカタログに、t1のメタ情報を追加する権限
- TOASTテーブルを作成する権限
- インデックスを作成する権限
- ➡ SE-PostgreSQLは徹底的にチェックを行なう

その他の諸機能

- PGACE (PostgreSQL Access Control Extension)
 - 他のセキュアOSでも、類似機能を実装する枠組み
 - Trusted Solaris開発者との議論の中で生まれた
- Trusted Procedure
 - 機密情報にアクセスするための安全な手段を提供
 - SELinuxのドメイン遷移の考え方を応用
- 拡張pg_dump/pg_dumpall
 - --enable-selinux オプションで、
セキュリティコンテキスト付きのバックアップ

SE-PostgreSQL利用環境



※ 接続元IPアドレス/ポートに基づくクライアント権限は、現在開発中の機能

The SE-PostgreSQL Security Guide

■ SE-PostgreSQLの公式ドキュメント

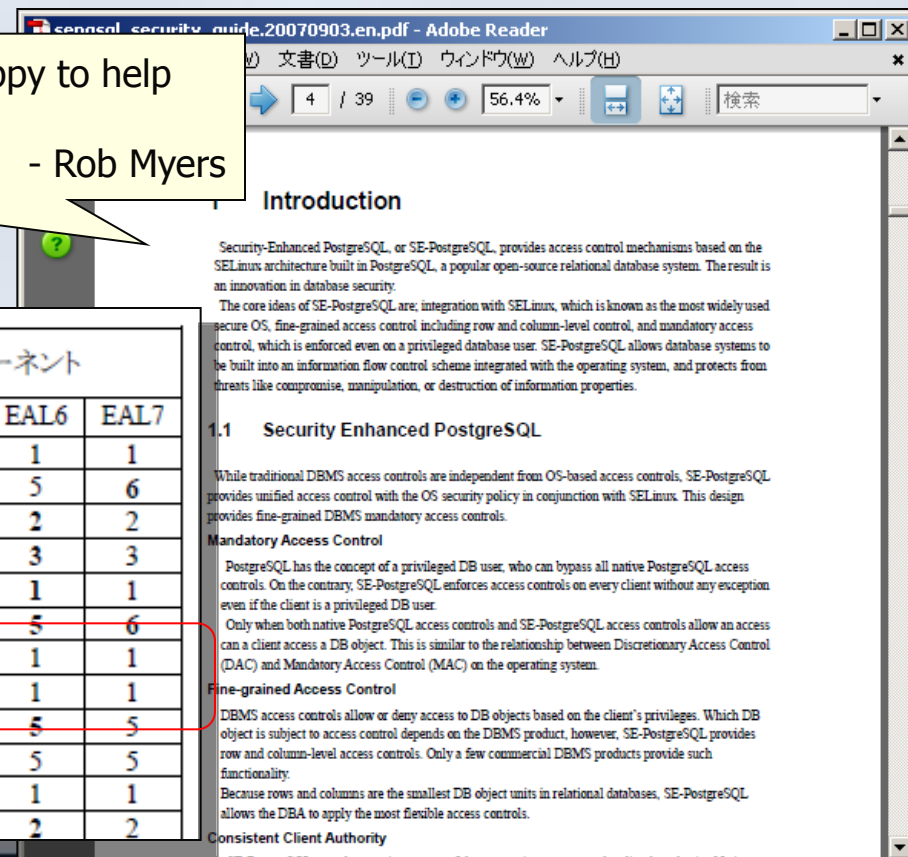
✓ 日本語版／英語版の両方を提供

I am a native English speaker, and would be happy to help improve the readability of the security guide.

- Rob Myers

ISO15408 保証コンポーネント

| 保証クラス | 保証ファミリ | 評価保証レベル別の保証コンポーネント | | | | | | |
|---------|---------|--------------------|------|------|------|------|------|------|
| | | EAL1 | EAL2 | EAL3 | EAL4 | EAL5 | EAL6 | EAL7 |
| 開発 | ADV ARC | | 1 | 1 | 1 | 1 | 1 | 1 |
| | ADV FSP | 1 | 2 | 3 | 4 | 5 | 5 | 6 |
| | ADV IMP | | | | 1 | 1 | 2 | 2 |
| | ADV INT | | | | | 2 | 3 | 3 |
| | ADV SPM | | | | | | 1 | 1 |
| ガイダンス文書 | AGD OPE | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | AGD PRE | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| ライフサイクル | ALC CMC | 1 | 2 | 3 | 4 | 4 | 5 | 5 |
| | ALC CMS | 1 | 2 | 3 | 4 | 5 | 5 | 5 |
| | ALC DEL | | 1 | 1 | 1 | 1 | 1 | 1 |
| | ALC DVS | | | 1 | 1 | 1 | 2 | 2 |

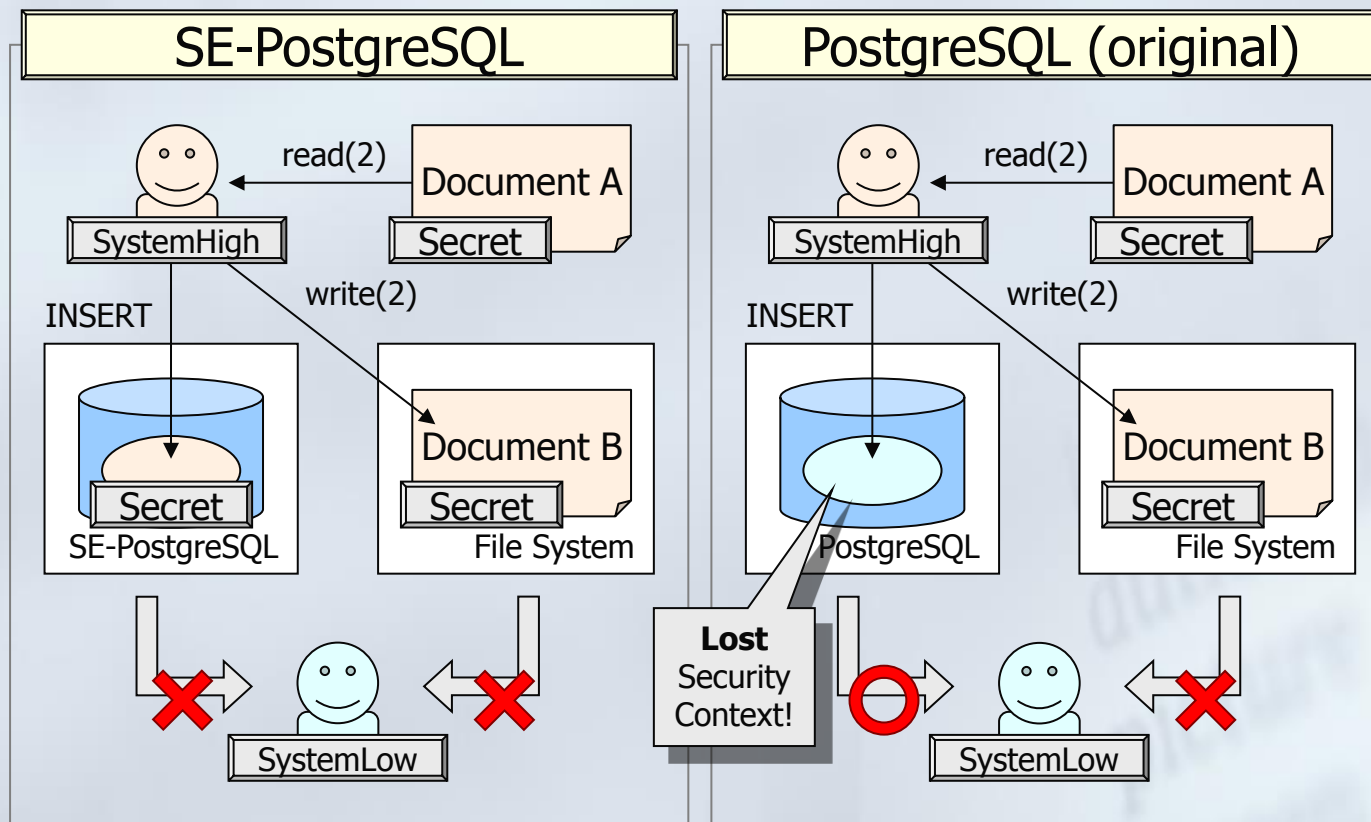


本日のアジェンダ

- I. プロジェクトの概要・背景・思想
- II. SE-PostgreSQLアクセス制御方式
- III. デモンストレーション
- IV. SE-PostgreSQLの展開と未来

デモンストレーション

- 行レベルアクセス制御の例
- オリジナルPostgreSQLとの違い



本日のアジェンダ

- I. プロジェクトの概要・背景・思想
- II. SE-PostgreSQLアクセス制御方式
- III. デモンストレーション
- IV. SE-PostgreSQLの展開と未来

SE-PostgreSQLの展開 (1)

■ 対外発表

- SELinux Symposium 2007/Baltimore
- PostgreSQL Conference 2007/Tokyo、他

■ SELinux Developer's Summit での議論

- LinuxカーネルのSE-PostgreSQL対応インターフェース
- 標準セキュリティポリシーのSE-PostgreSQL対応

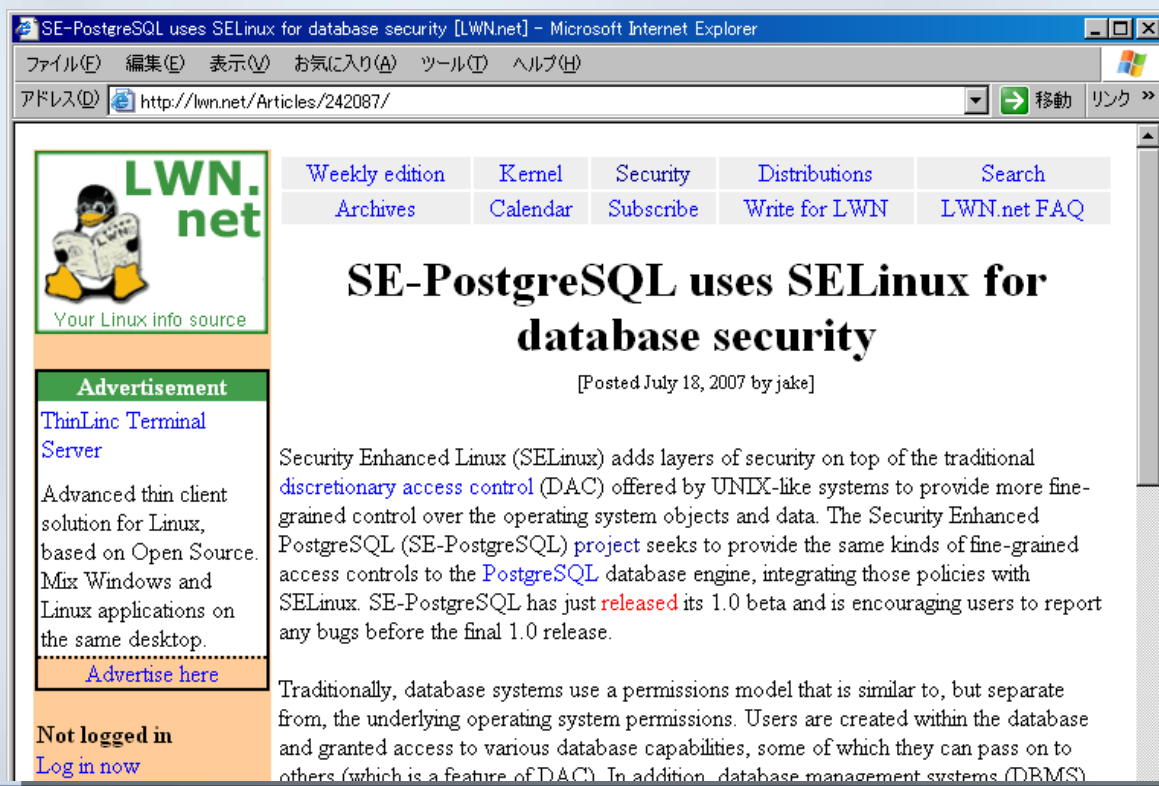


SE-PostgreSQLの展開 (2)

■ @ITでの連載記事

「SE-PostgreSQLによるセキュア・データベース構築」

■ “Linux Weekly News” で取り上げられる



SE-PostgreSQLの展開 (3)

■ Fedora Project

- Fedora開発版(rawhide)にSE-PostgreSQLがマージ
- ✓ selinux-policy-3.0.6以降のSE-PostgreSQL対応
- ✓ Fedora Packaging Guideline に基づくレビュー
 - ➡ Fedora 8 以降での利用が可能に

■ PostgreSQLコミュニティ

- Trusted Solaris開発者との議論
 - 類似機能の実装を容易にする PGACE
PGACE: PostgreSQL Access Control Extension
- PostgreSQL本流への統合
 - 現在は PostgreSQL 8.3 への Features Freeze 中
 - ターゲットは PostgreSQL 8.4 (2008秋?)

SE-PostgreSQLの情報源

■ 公式サイト

- <http://code.google.com/p/sepgsql/>
 - ✓ ドキュメント、RPMパッケージ、Subversionリポジトリ

■ メーリングリスト

- sepgsql@kaigai.gr.jp

■ @IT 連載記事

- SE-PostgreSQLによるセキュア・データベース構築
<http://www.atmarkit.co.jp/fsecurity/rensai/sepgsql01/sepgsql01.html>

■ Fedora Project

- 各ミラーサイトより、開発版(rawhide)を取得可能

今後の展望

■ 対 Fedora Project

- Fedoraのパッケージとして、実績を蓄積すること。
- 将来的には商用ディストロへも...

■ 対 PostgreSQLコミュニティ

- PostgreSQL 8.4.x (2008秋) に向けて議論を開始
- PGACE/SE-PostgreSQL機能のプッシュ

■ 次バージョンのSE-PostgreSQLへ向けて

- pl/pgSQLスクリプト言語拡張
- Polyinstantiationテーブル
- etc, ...

最後に

- SE-PostgreSQL開発のポイント
 - ➡ オープンソースコミュニティとの連携
 - SELinux, PostgreSQL, Fedora Project, etc...
- その分野の専門家の知見をフィードバックできる
 - ➡ ソフトウェア品質の向上、新機能の提案
- 関連するパッケージの対応
 - Linuxカーネル、selinux-policy、libselenium、etc...
 - ➡ より多くのユーザが利用することに
- つまり、Powered by OSS Community ! !



Any question?



Thank you!