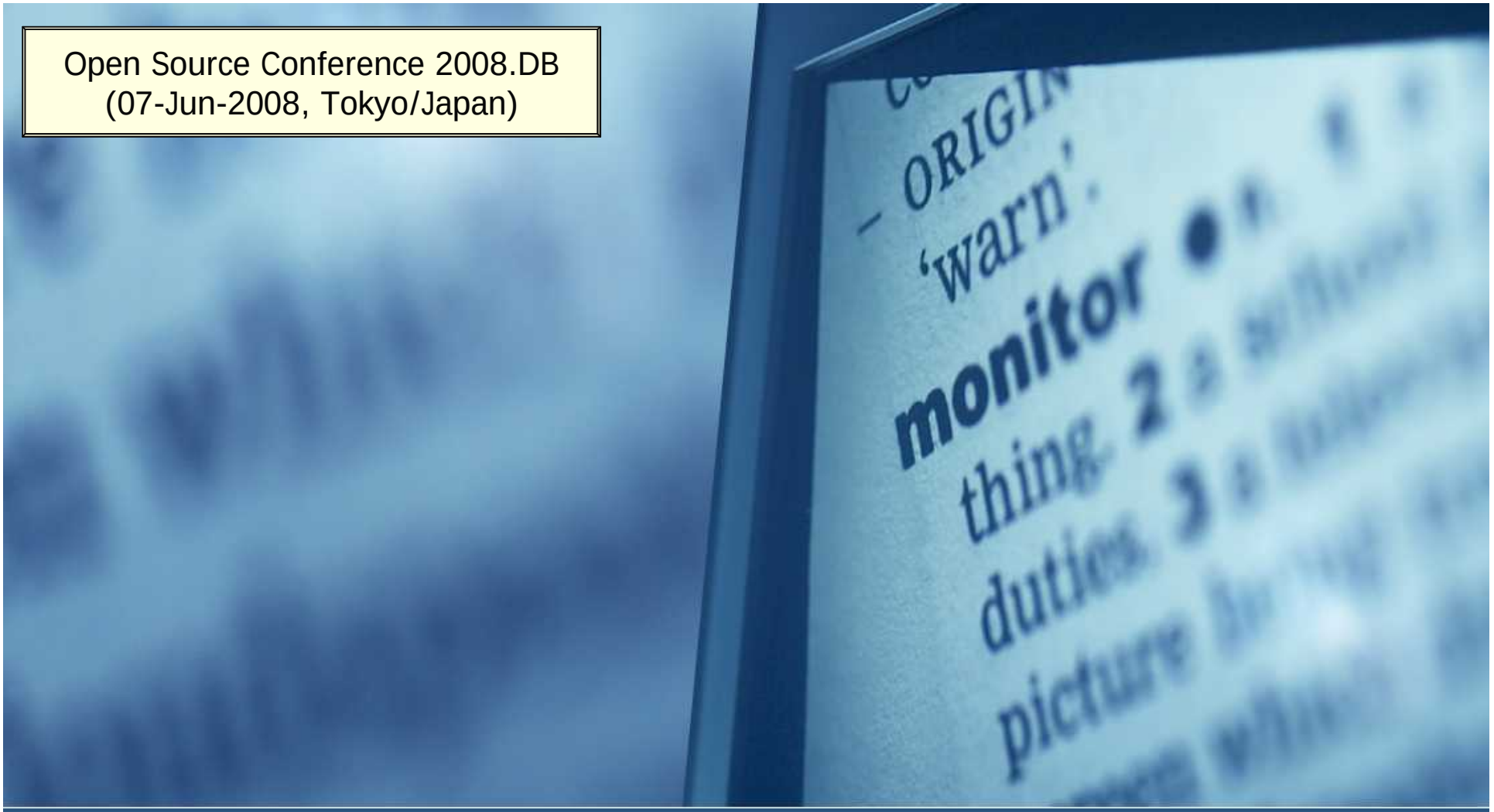


Open Source Conference 2008.DB
(07-Jun-2008, Tokyo/Japan)

Security-Enhanced PostgreSQL

日本セキュアOSユーザ会　海外浩平 <kaigai@kaigai.gr.jp>

はじめに

- 本日の資料
  - SE-PostgreSQLプロジェクト：
    - http://code.google.com/p/sepgsql/
    - http://sepgsql.googlecode.com/files/OSC2008.DB-sepgsql.pdf
- 自己紹介
  - 所属
    - ＮＥＣ　ＯＳＳプラットフォーム開発本部
    - 日本セキュアＯＳユーザ会
  - Linuxカーネル開発に従事
    - 特にSELinux、セキュリティ関連分野
  - Security-Enhanced PostgreSQL
    - 2006年夏から開発を開始
    - IPA 未踏ソフトウェア創造事業

思想的背景

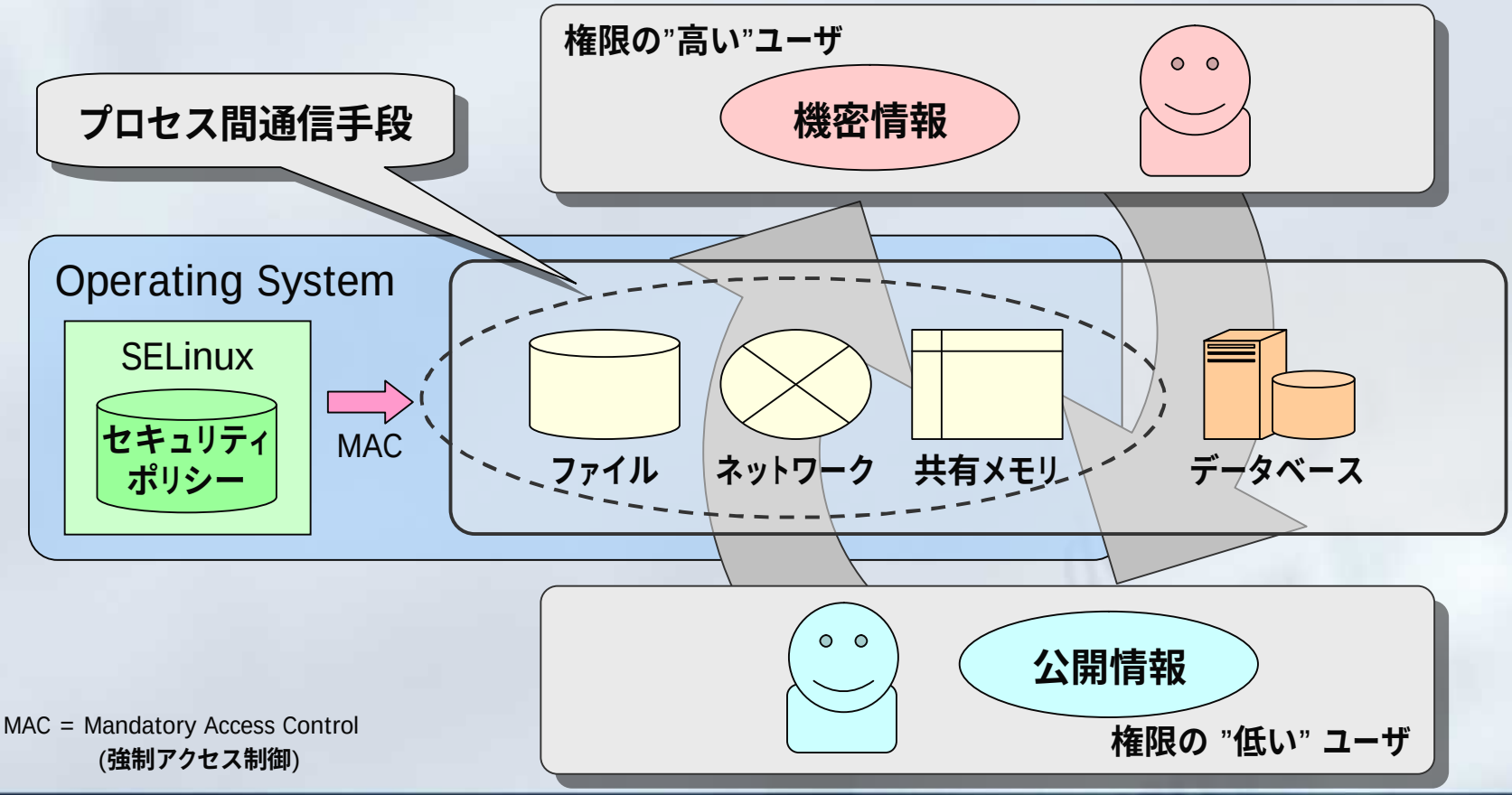
手帳の価値	¥1,280
手帳にメモした内容	¥PRICELESS

- 我々が守りたいもの
  - 個人情報、企業秘密、認証データ、etc...
  - 無形の “情報資産”
- 中身は同じ、アクセス制御は・・・？
  - ファイル ... UNIX permission
  - データベース ... Database ACL
  - “情報資産” を格納する “手段” に強く依存

同一の ”情報資産” に対しては、同一のアクセス制御ポリシーを適用するべき。
断じて、それを格納する ”手段” は主役ではない。

アクセス制御の一貫性

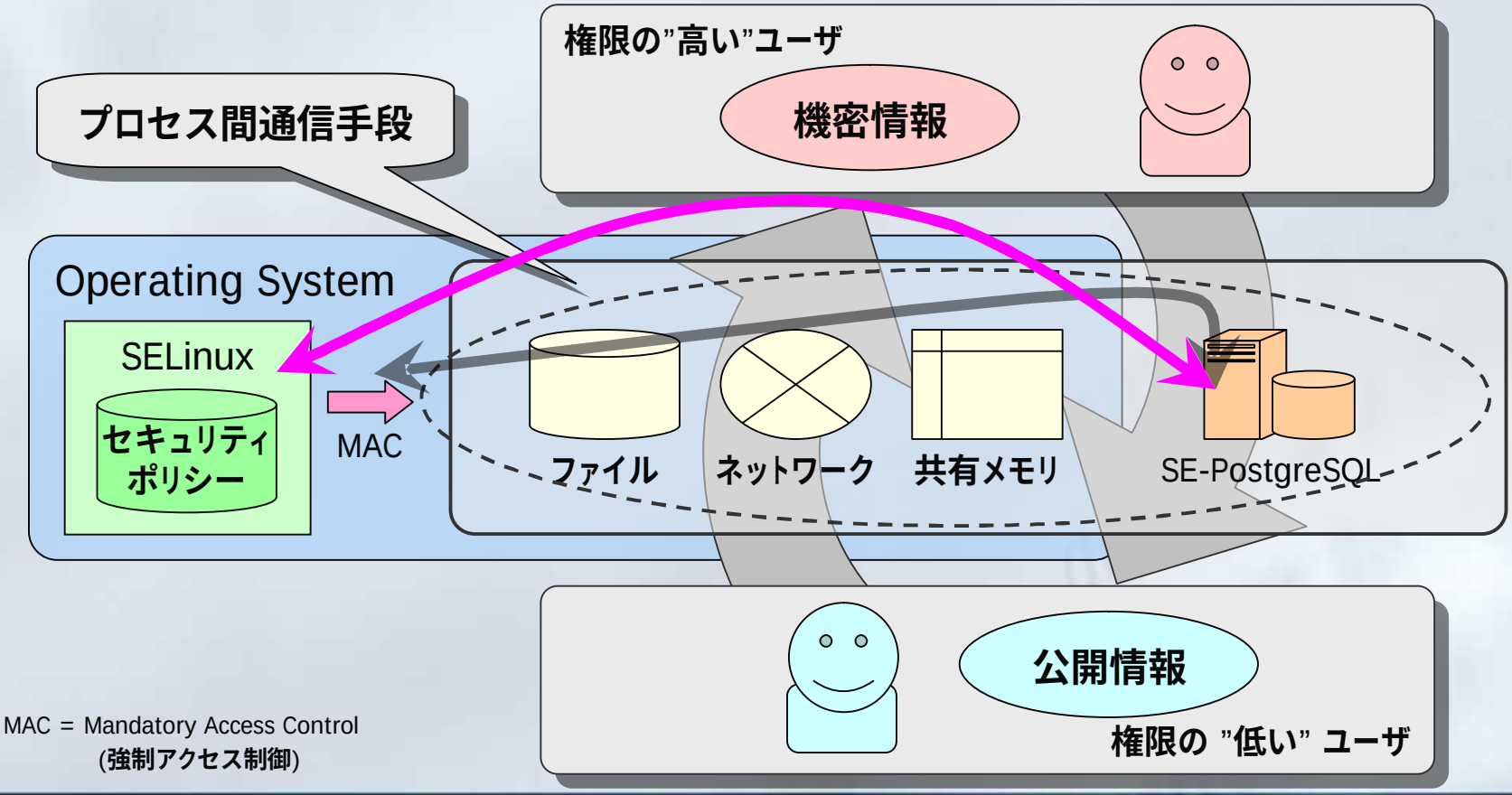
- 経路に関係なく、機密情報が漏洩してはならない。
- 例外なしに、全てのユーザ・全てのオブジェクトに適用する。



MAC = Mandatory Access Control
(強制アクセス制御)

アクセス制御の一貫性

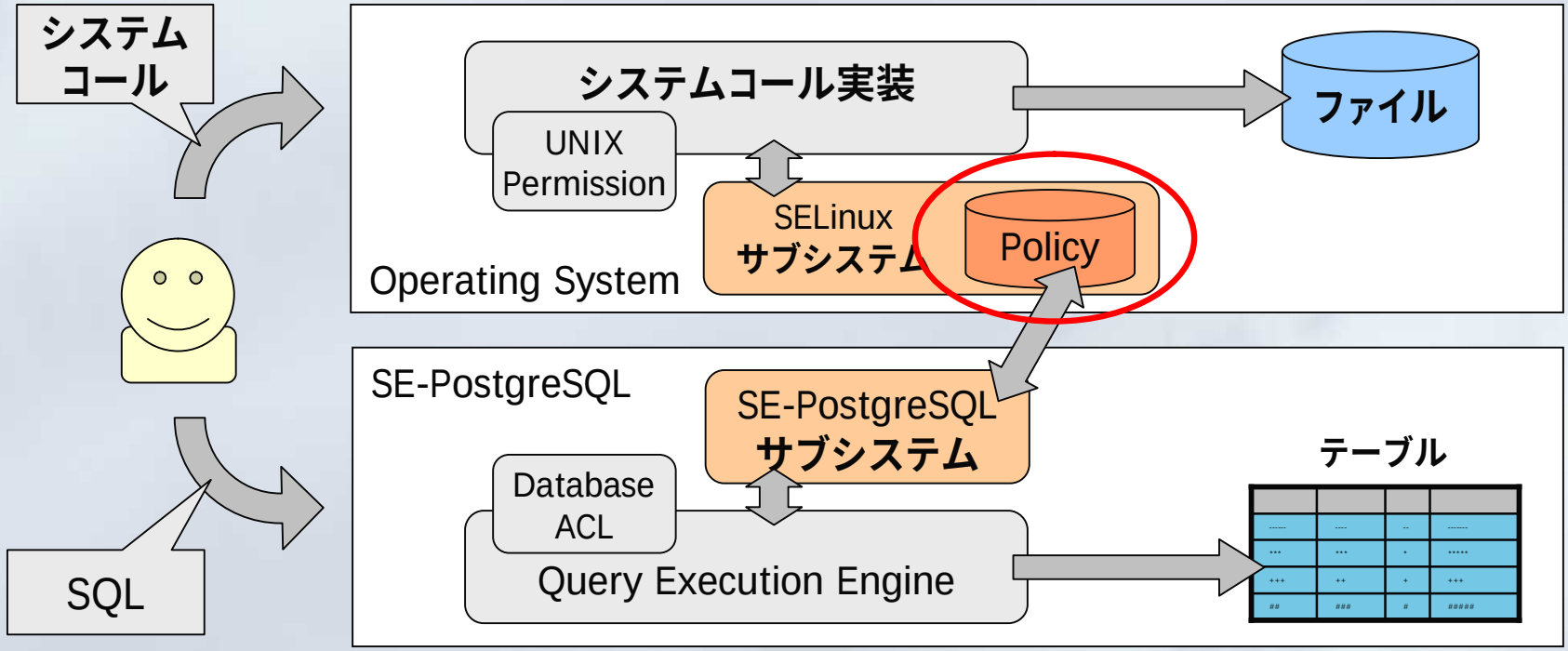
- 経路に関係なく、機密情報が漏洩してはならない。
- 例外なしに、全てのユーザ、全てのオブジェクトに適用する。



MAC = Mandatory Access Control
(強制アクセス制御)

OS/DBMSの類似性

- OS　　… プロセス、システムコール、ファイルシステム
- DBMS　　… クライアント、SQL、DBオブジェクト
- SE-PostgreSQLの目標
  - あたかも、ファイルシステムを参照しているかのようなアクセス制御



SE-PostgreSQLの特徴

- アクセス制御における“システムワイド”な一貫性
  - 単一のセキュリティポリシーをOSと共有
  - OSのアクセス制御と決して矛盾しない
- 細粒度・強制アクセス制御
  - 行レベル／列レベルアクセス制御を含む
  - 特権ユーザを含む全てのユーザに適用
- ご利益
  - 情報フロー制御
  - 内部犯による情報漏えい／改ざんの防止
  - SQLインジェクション／アプリバグの被害最小化

共通のセキュリティ属性

- security_context システム列

```
postgres=# select security_context, * from drink;
                  security_context                  | id | name  | price
----------------------------------------------------+----+-------+-------
 unconfined_u:object_r:sepgsql_table_t              |  1 | water |   110
 unconfined_u:object_r:sepgsql_table_t              |  2 | coke  |   120
 unconfined_u:object_r:sepgsql_table_t              |  3 | milk  |   150
 unconfined_u:object_r:sepgsql_table_t              |  4 | juice |   130
 unconfined_u:object_r:sepgsql_table_t:Classified   |  5 | beer  |   240
 unconfined_u:object_r:sepgsql_table_t:Classified   |  6 | wine  |   380
(6 rows)
```

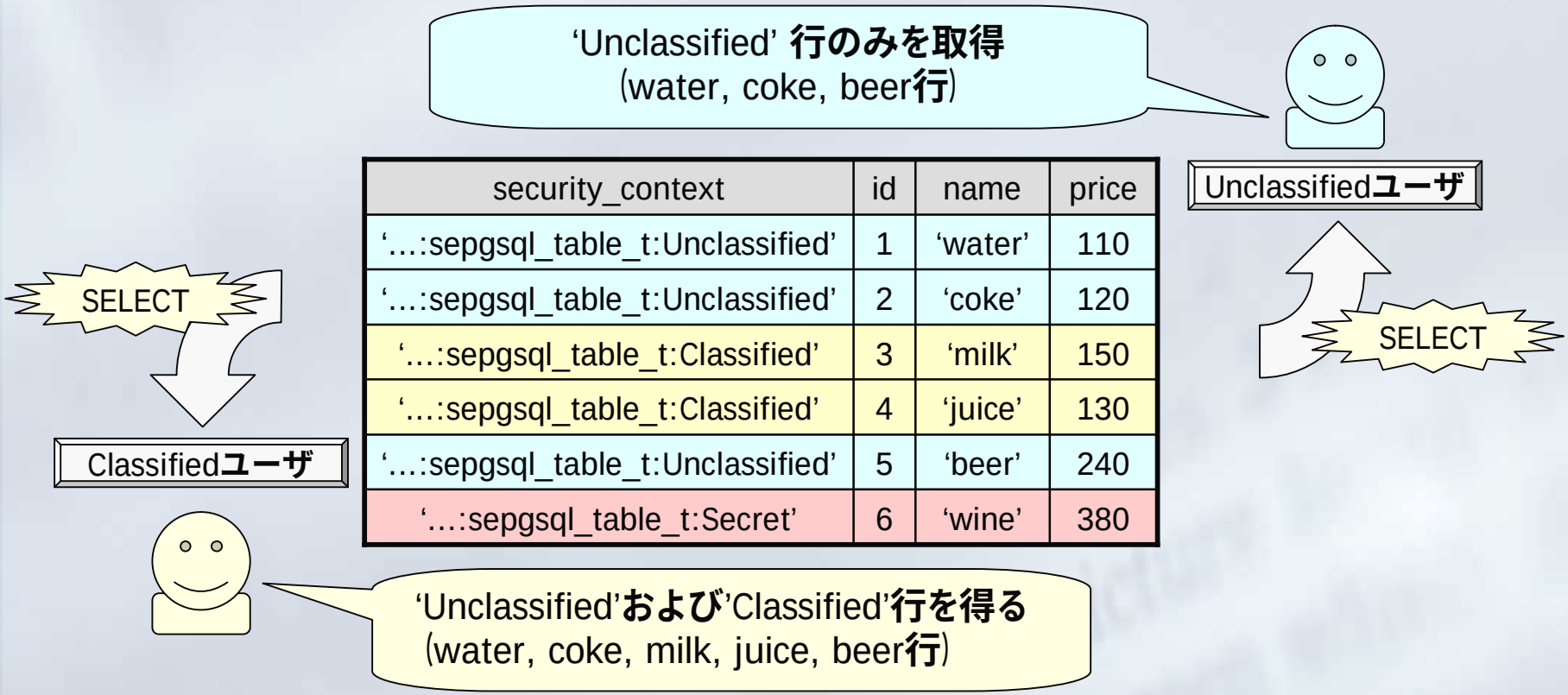
SELinuxが
アクセス制御に利用する

- ファイルシステムの場合

```
[kaigai@masu ~]$ ls -Z /etc/
-rw-r--r--  root root system_u:object_r:etc_aliases_t  aliases
-rw-r--r--  root root system_u:object_r:etc_t          auto.master
-rw-r--r--  root root system_u:object_r:etc_t          auto.misc
-rw-r--r--  root root system_u:object_r:etc_t          group
-r--------  root root system_u:object_r:shadow_t       shadow
-rw-r--r--  root root system_u:object_r:etc_t          passwd
    :         :    :              :                      :
```

行レベルアクセス制御

- アクセス権のない行は、”あたかも存在しないように” 扱われる
  - SELECT文 … 結果セットから除去
  - UPDATE/DELETE構文 … 更新/削除の対象から除去

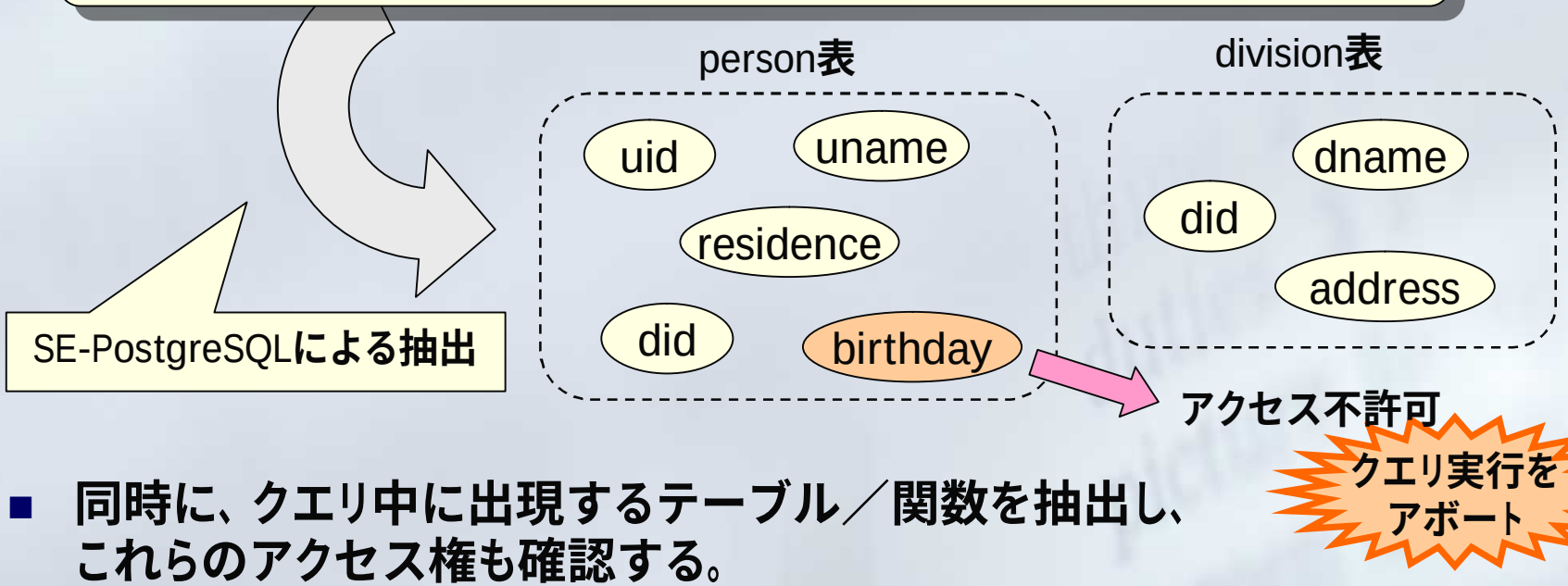


security_context	id	name	price
‘…:sepgsql_table_t:Unclassified’	1	‘water’	110
‘…:sepgsql_table_t:Unclassified’	2	‘coke’	120
‘…:sepgsql_table_t:Classified’	3	‘milk’	150
‘…:sepgsql_table_t:Classified’	4	‘juice’	130
‘…:sepgsql_table_t:Unclassified’	5	‘beer’	240
‘…:sepgsql_table_t:Secret’	6	‘wine’	380

列レベルアクセス制御

- アクセス権のない列への参照 → クエリの実行をアボート
- クエリ中に出現する全てのカラムをチェックする

```
SELECT uid, uname || '様', age(birthday), dname
FROM person p JOIN division d ON p.did = d.did
WHERE p.residence = '東京' OR d.address = '神奈川';
```



- 同時に、クエリ中に出現するテーブル／関数を抽出し、これらのアクセス権も確認する。

Case Study (1/2)

```
SELECT name, price * 2 FROM drink WHERE id < 40;
```

- db_column:{select} … **name列、price列**
- db_column:{use} … **id列**
  - {use} **パーミッション: 読出しを行うが、ユーザには返さない**
- db_procedure:{execute} … **int4mul関数、int4lt関数**
- db_table:{select use} … **drink表**
  - **ポリシーに違反する場合、SQLクエリの実行を中止し、トランザクションをアボートする**

演算子の実装

および

- db_tuple:{select use} … **各タプル**
  - **アクセス権のないタプルは、結果セットから除去される。**

Case Study (2/2)

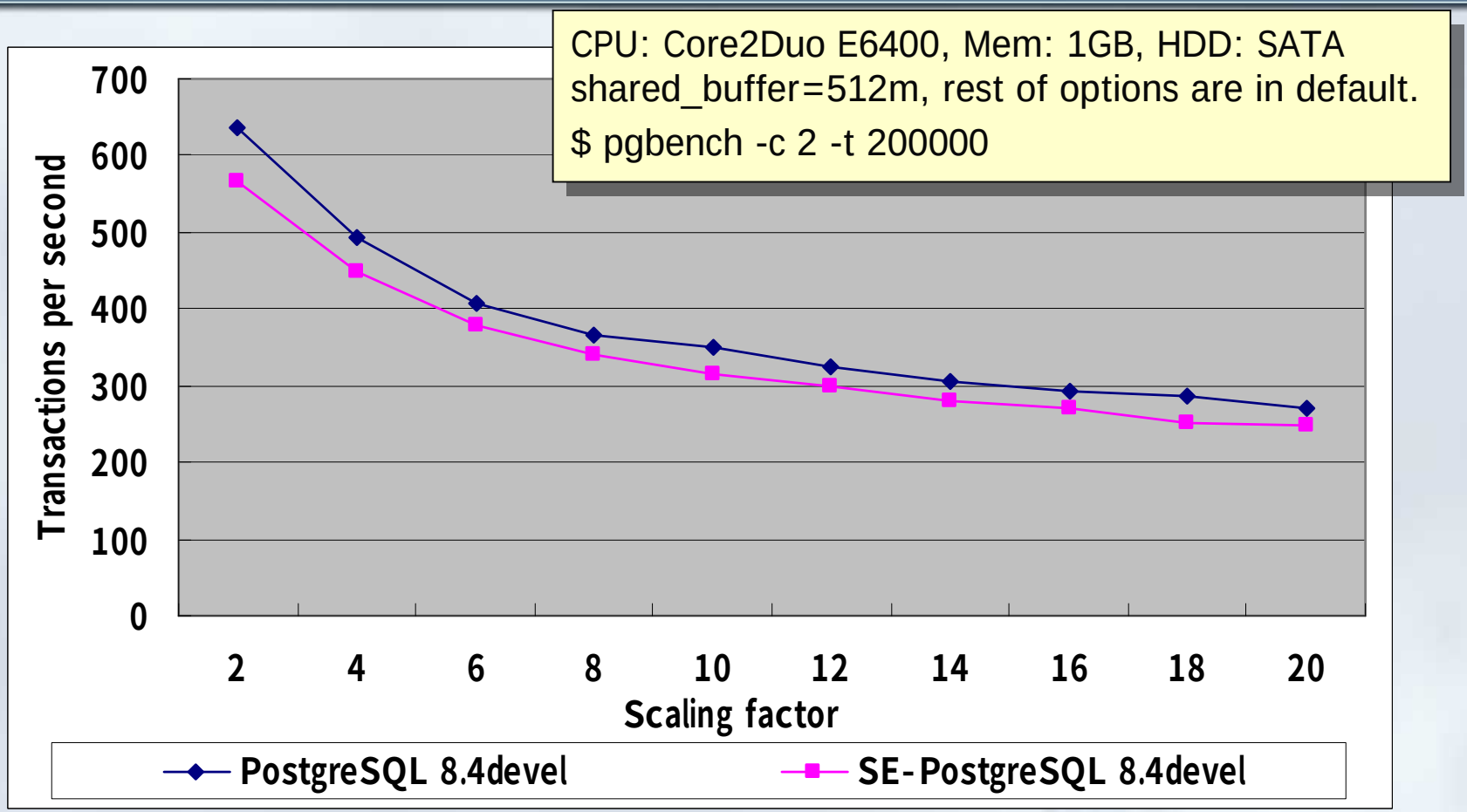
```
UPDATE drink SET size = 500, price = price * 2
                 WHERE alcohol = true;
```

- db_column:{update} ... **size列**
- db_column:{select update} ... **price列**
  - **price列は更新と同時に、読出しも行われる**
- db_column:{use} ... **alcohol列**
- db_procedure:{execute} ... **booleq関数、int4mul関数**
- db_table:{select use update} ... **drink表**
  - **ポリシーに違反する場合、SQLクエリの実行を中止し、トランザクションをアボートする**

および、

- db_tuple:{select use update} ... **各タプル**
  - **アクセス権のないタプルは、更新の対象から除外される。**

Performance



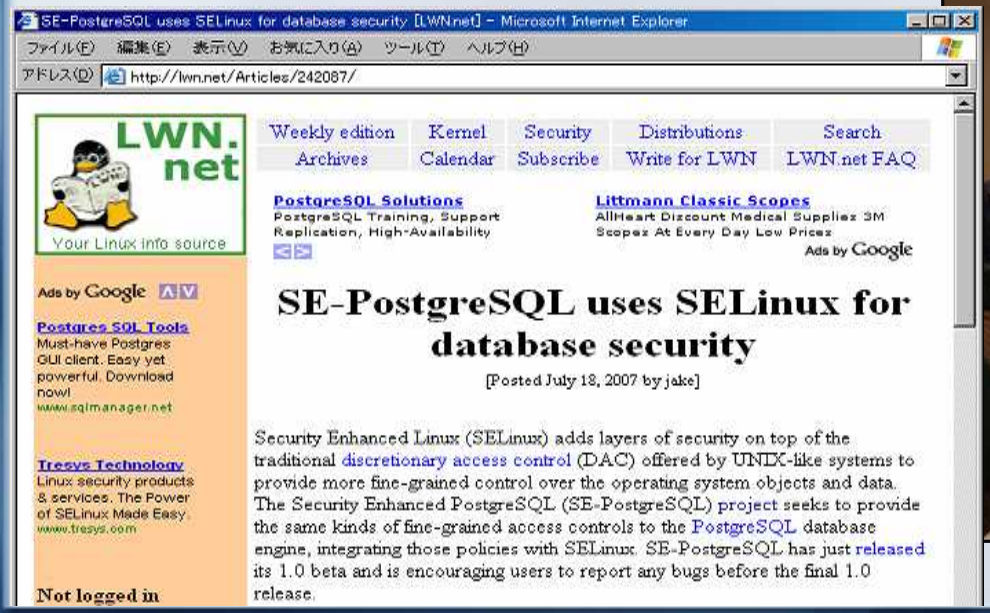
- **約10%弱のトレードオフ**
- access vector cache (AVC): **カーネル呼び出し回数最小化**

Demonstration

- 行レベルアクセス制御
  - ユーザの権限に応じて、結果セットが変わります。
- 列レベルアクセス制御
  - ユーザの権限次第では、参照できない列があります。
  - しかし、All or Nothing以外のアクセス方法も可能です。

World Wideでの展開 (1/3)

- 対SELinuxコミュニティでの活動
  - アクセス制御モデルの設計
  - Linux kernel 機能の拡張
  - 標準セキュリティポリシーへの統合
- Fedora Projectでの採用
  - Fedora 8 以降～

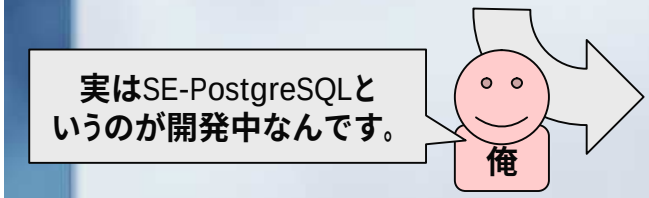



SELinux Developer Summit 2007
@Baltimore, USA

World Wideでの展開 (2/3)

- PostgreSQLコミュニティでは…。

Data: 2007-03-03
From: Josh Berkus
I'm chasing a rumor that someone is working on integrating PostgreSQL with the SELinux security framework. Anyone know anything about this?



Data: 2007-03-05
From: KaiGai Kohei
Subject: [ANN] SE-PostgreSQL 8.2.3-1.0 alpha release

PostgreSQL version 8.3.0 Feature Freeze (’08/04/01)

Data: 2007-04-17
From: KaiGai Kohei
Subject: [RFC] PostgreSQL Access Control Extension (PGACE)

Data: 2007-04-19
From: Tome Lane
Well, personally I won't have any cycles to think hard about any post-8.3 work until after the beta is out.

orz

World Wideでの展開 (3/3)

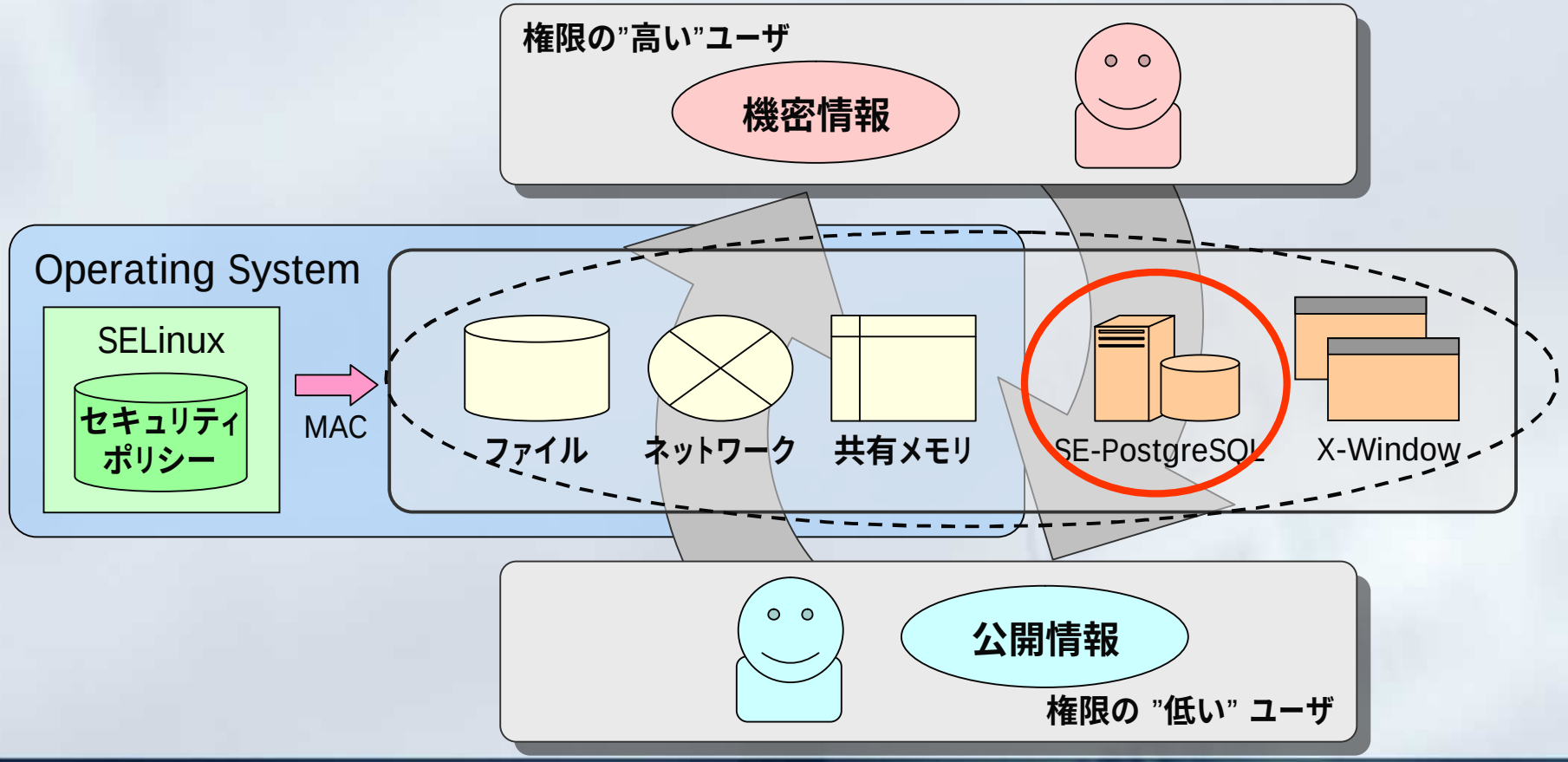
- PostgreSQLコミュニティでの活動...その後
  - v8.4開発サイクル向けに CommitFest:May にパッチを投稿
    - 物凄い勢いでフィードバック、応援メッセージも
  - PGcon2008@Ottawaでの発表

PGcon 2008@Ottawaでの発表の様子

Date: 2008-05-01
From: Josh Berkus
Folks, For hackers who don't understand security frameworks, I'm going to make a strong case for KaiGai's patch. Because of ....

SE-PostgreSQLの今後

- PostgreSQL v8.4 での統合！
- 必要なことは何でもやる。



情報源

- SE-PostgreSQL Home

  http://code.google.com/p/sepgsql/

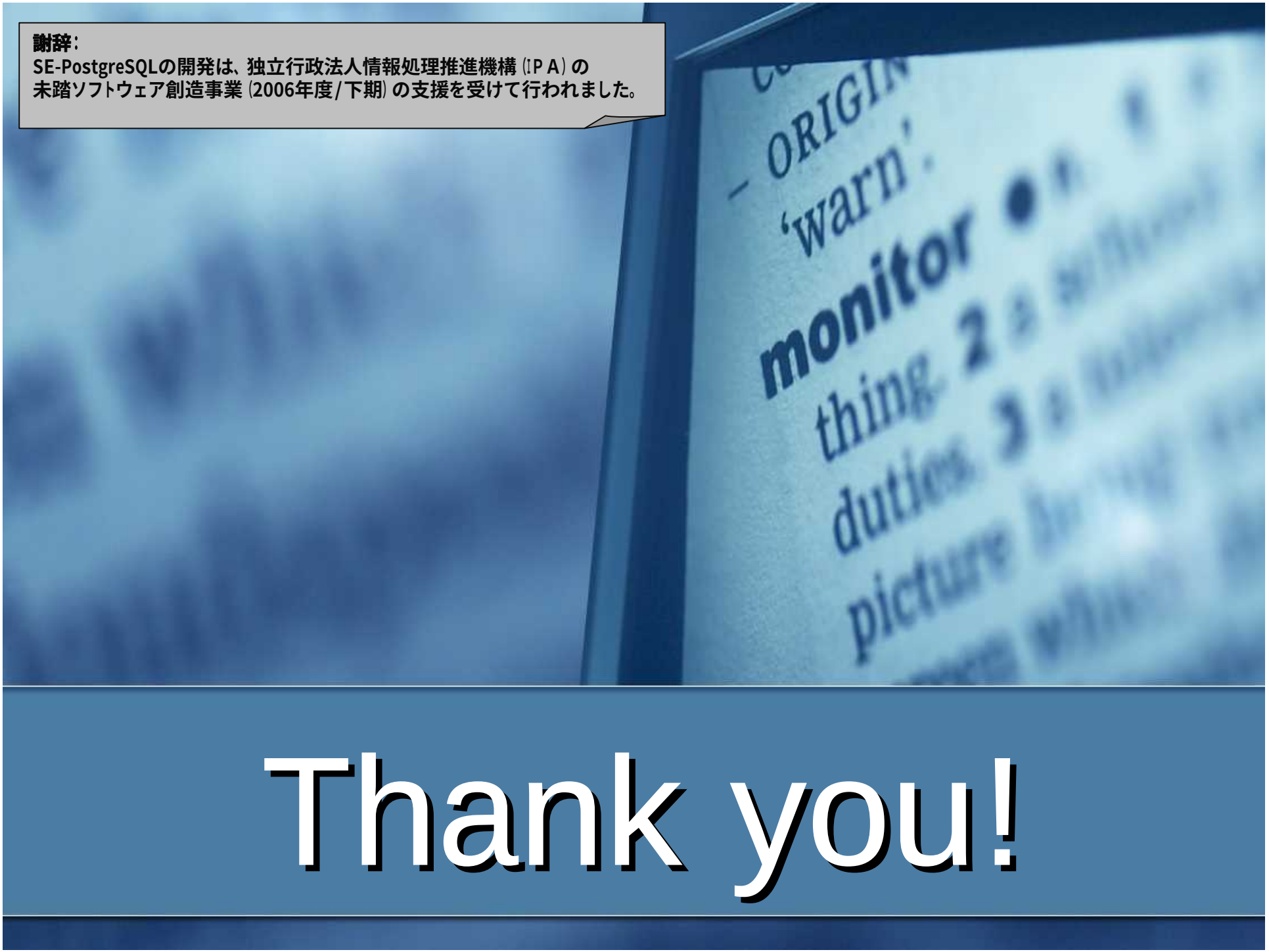
  - SVNリポジトリ、RPMパッケージ
  - The SE-PostgreSQL Security Guide (日本語/英語)
- 日本セキュアOSユーザ会

  http://www.selinux.gr.jp/ml.html

  - メーリングリスト紹介
- @IT『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

Any Question?


謝辞:
SE-PostgreSQLの開発は、独立行政法人情報処理推進機構（IPA）の
未踏ソフトウェア創造事業（2006年度/下期）の支援を受けて行われました。
ORIGIN
'warn'
monitor
thing 2
duties 3
picture
Thank you!