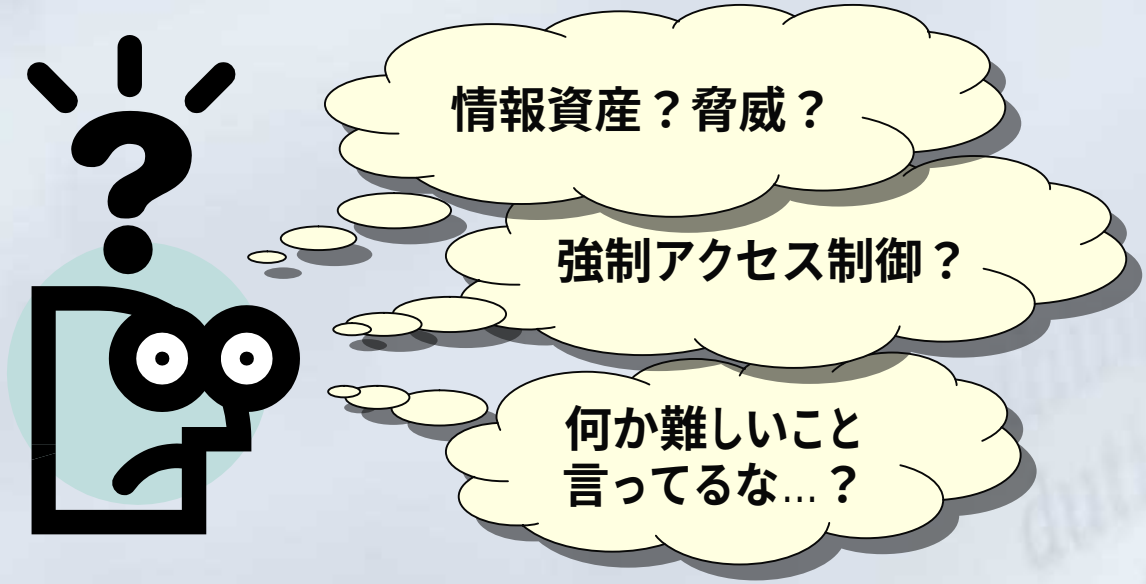
もうSELinuxは怖くない！

～セキュアLinux徹底解説～

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

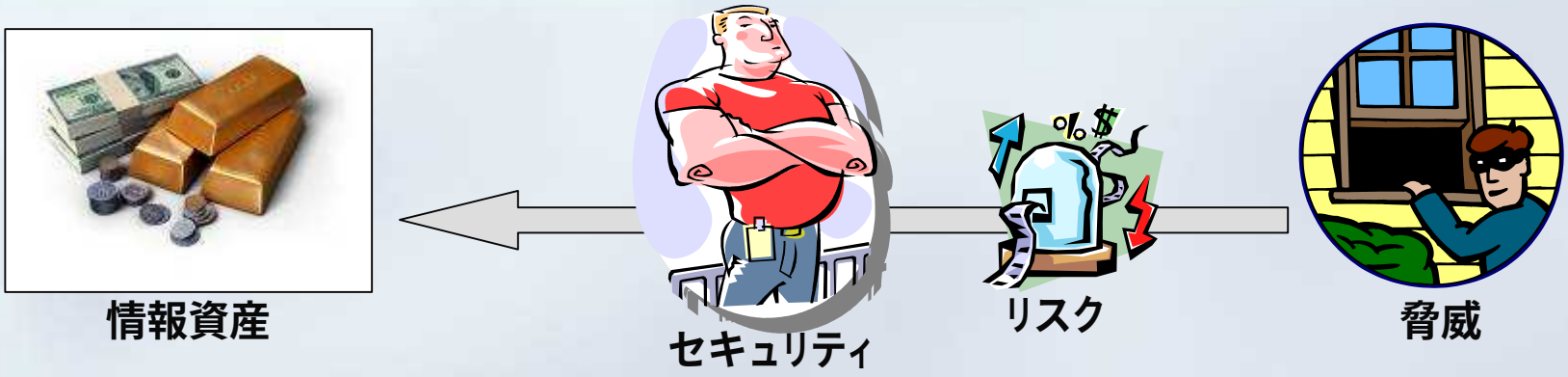
SELinuxって何でしょう？

- **強制アクセス制御**により、システムに格納された**情報資産**を、漏えい／改ざんといった**脅威**から保護する Linux カーネルの機能。



- 急がば回れ
  - 先ずは、背景にあるセキュリティの思想から理解しましょう。

“セキュリティ”とは

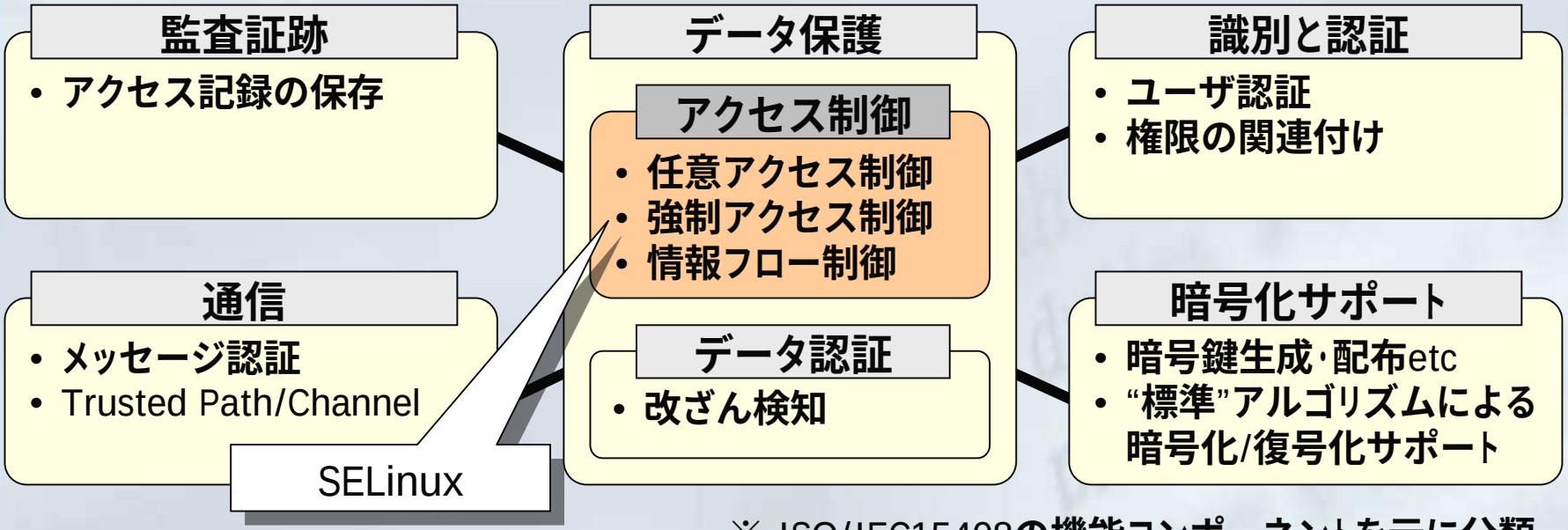


- 情報資産　＝ 価値あるもの／守るべき対象
- 脅威　＝ 情報資産の下記特性が満たされなくなる事
  - 機密性 … 適切な権限のない利用者に情報が漏えいする事
  - 完全性 … 適切な権限のない利用者が情報を改ざんする事
  - 可用性 … 必要な時に、必要な情報を利用する事ができる事
- リスク　＝ 脅威が顕在化する可能性

セキュリティ ＝ リスクを減少させるための手段

セキュリティ対策

- 非技術的対策
  - システム運用方針、システム利用環境、従業員教育、etc...
- 技術的対策
  - アクセス制御、認証、暗号化、etc...

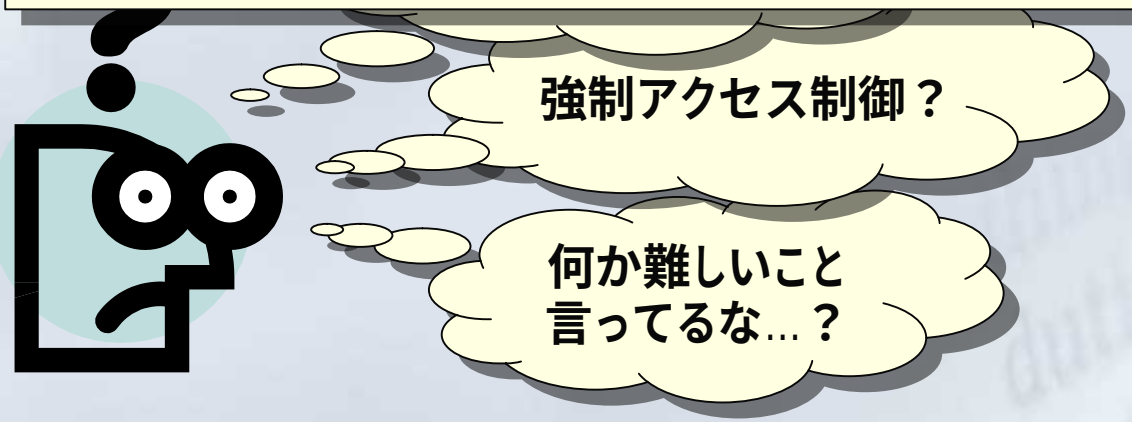


※ ISO/IEC15408の機能コンポーネントを元に分類

SELinuxって何でしょう？

- 強制アクセス制御により、システムに格納された情報資産を、漏えい／改ざんといった脅威から保護するカーネルの機能

SELinuxの提供する強制アクセス制御とは何か？
なぜ、それが必要なのか？



- 急がば回れ
  - 先ずは、背景にあるセキュリティの思想から理解しましょう。

2種類のアクセス制御

- **任意アクセス制御** (DAC: Discretional Access Control)
  - 伝統的 UNIX アクセス制御モデル
  - 資源(ファイル)の所有者は、任意にアクセス権を変更できる

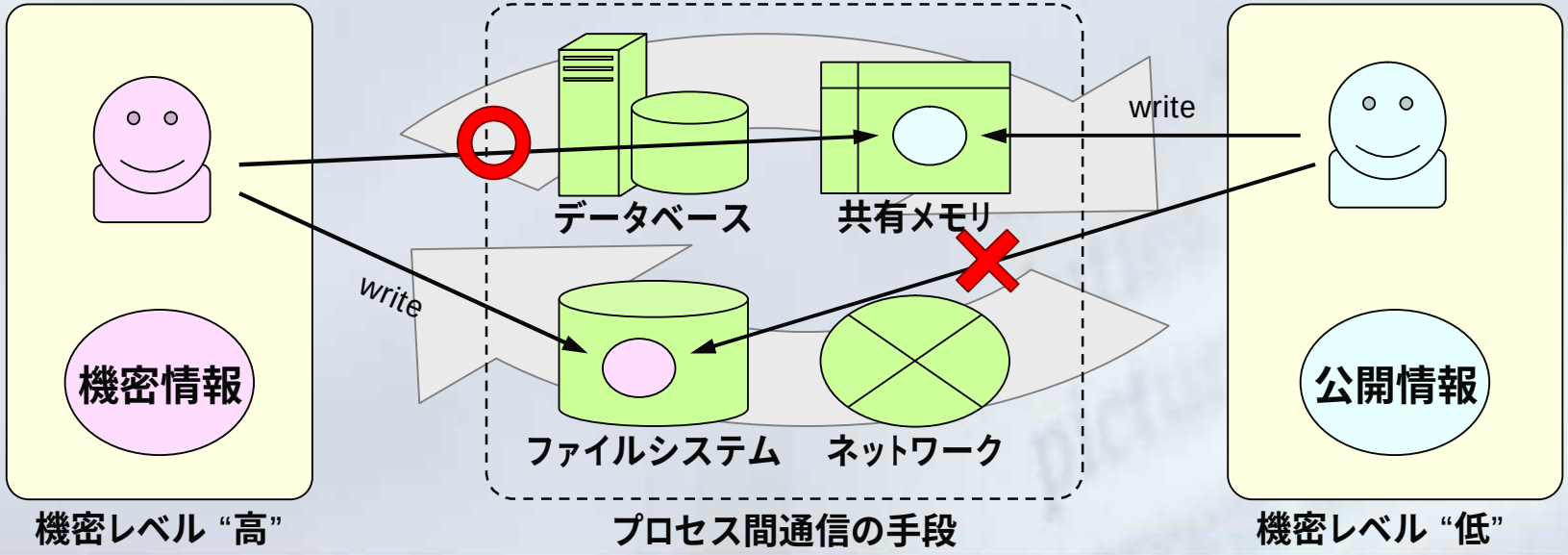
```
$ chmod o+r ~/my_secret_file.txt
```

  - root ユーザには、“任意アクセス制御” を無視できる “特権”
- **強制アクセス制御** (MAC: Mandatory Access Control)
  - SELinuxのアクセス制御モデル
  - 資源の所有者であっても、アクセス権を (任意には) 変更できない
    - セキュリティポリシーにより、アクセス権は “強制的に” 設定される。

- なぜ？ アクセス権を “任意に” 設定できてはいけないのか？
- なぜ？ root ユーザの権限を縛らなくてはならないのか？

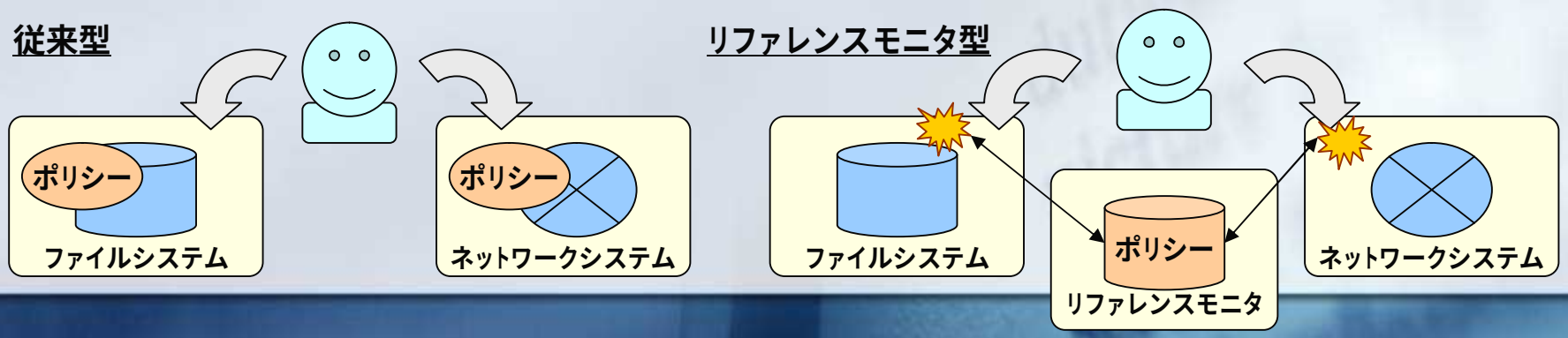
情報フロー制御

- “情報” は無形の資産
  - 何かに格納される／何かを通じて他の利用者に送信される
- 格納手段／通信手段とは独立な、一貫したアクセス制御が必要
- アクセス権を “任意に” 設定されては困る
  - 勝手に “機密情報” を公開扱いにされちゃったら？
  - アクセス制御の “例外扱い” なユーザが存在しちゃったら？



リファレンスモニタ

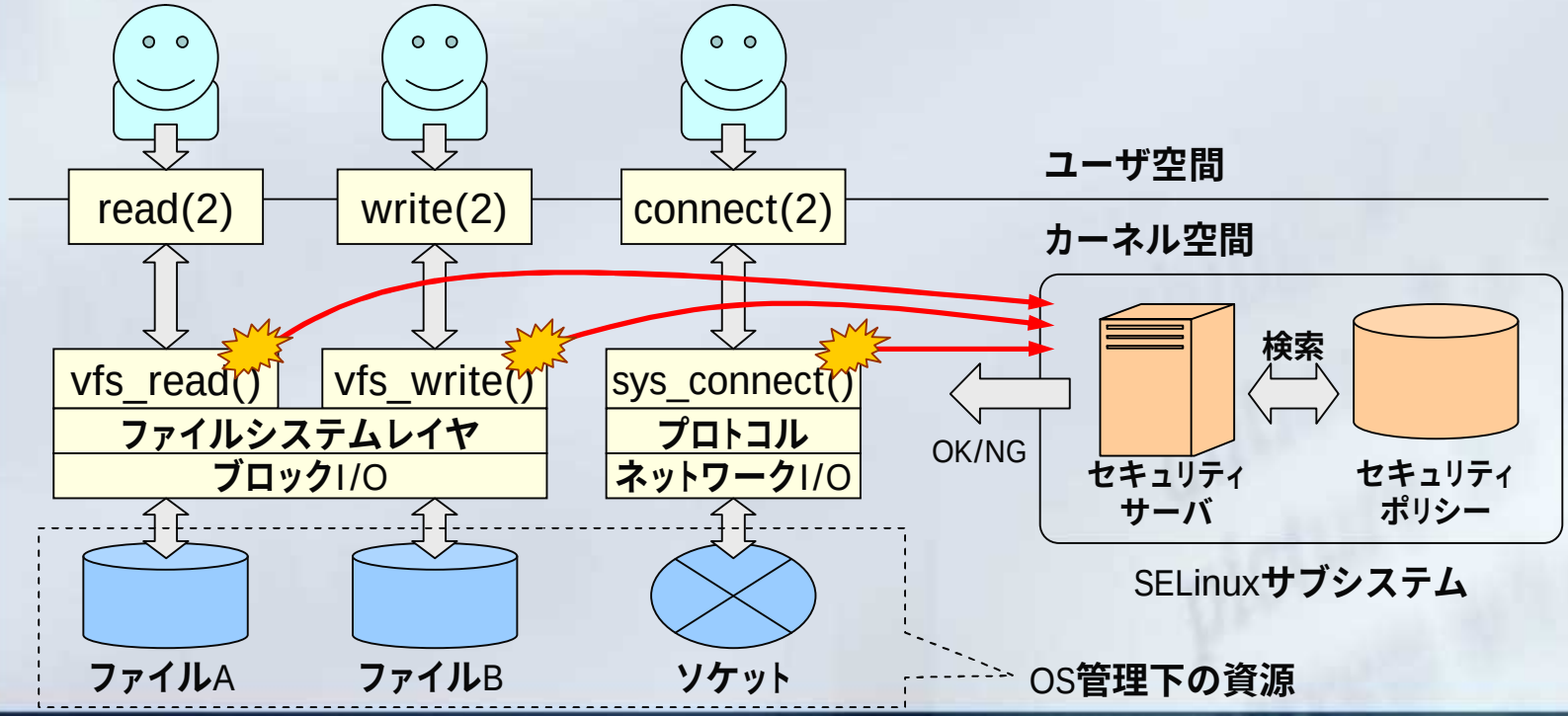
- リファレンスモニタとは
  - 利用者がシステム資源にアクセスする際に、これを漏れなく捕捉し、その可否をセキュリティポリシーに従って一元的に決定するモジュール
  - 情報フロー制御の実現のために必要なアイデア
    - アクセス制御の"漏れ"となるプロセス間通信手段が存在してはならない
    - アクセス制御の意思決定が、プロセス間通信手段に依存してはならない
  - リファレンスモニタの3要件
    - Always Invoked … 資源へのアクセスの度に、常にチェックが行なわれる
    - Tamper Proof … それ自身が改ざんから保護されている
    - Small Enough … それ自身のバグを検証可能である



じゃ、SELinuxとの関係はどうなのよ？

SELinuxとは

- Linuxにおけるリファレンスモニタの実装
  - ”資源”のアクセスには、”システムコール” 呼出が必要
  - “システムコール” 呼出をフックすれば、全ての資源に対するアクセスを捕捉できる

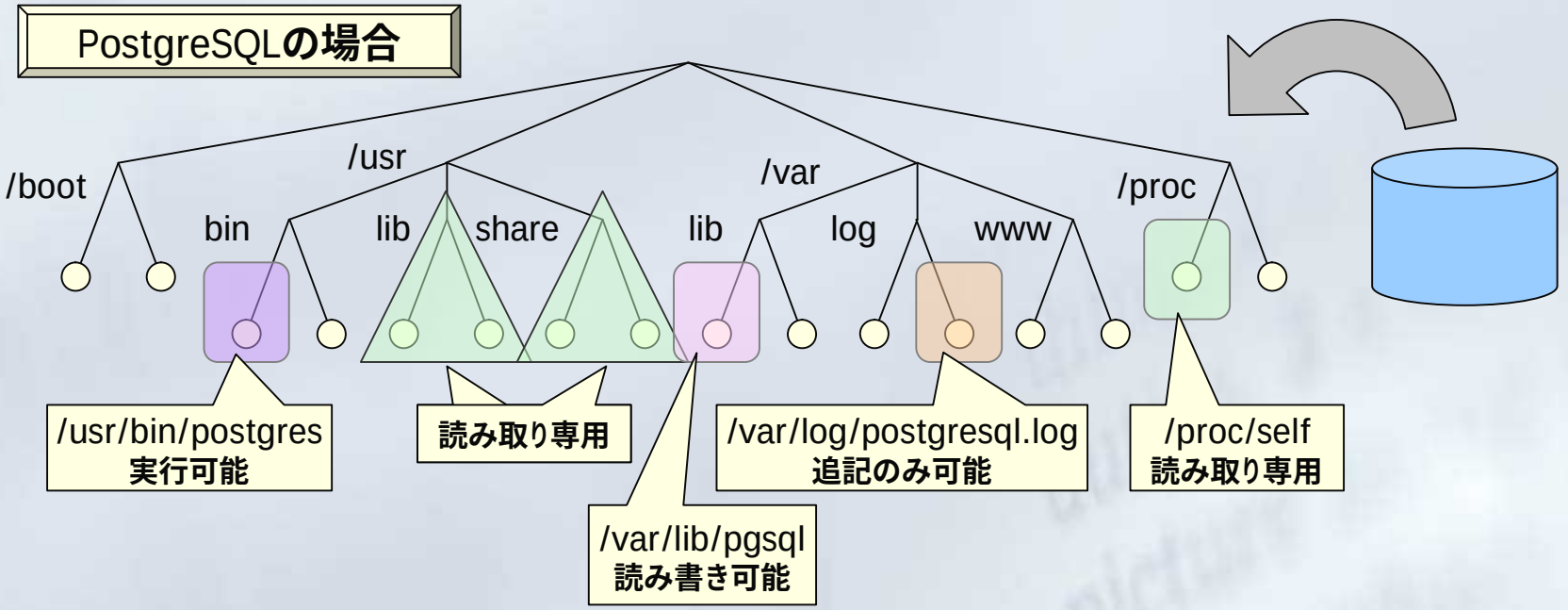


SELinuxのアクセス制御方式

- じゃあ、どうやって白黒つけるの？
- アクセス制御モデル
  - TE(Type Enforcement)
    - プログラムのアクセス可能なリソースの範囲に”枠”を作る
    - 操作対象は”枠”の内側か？外側か？
  - MLS(Multi Level Security)/MCS(Multi Category Security)
    - 上下関係を持つ”機密レベル”
    - 包含関係を持つ”機密区分”
    - この2つの制約条件を満たすか、否か？
  - RBAC(Role Based Access Control)
    - TEと密接に関連した方式で、難易度は高め。
    - 今回は説明から外します…。

Type Enforcement

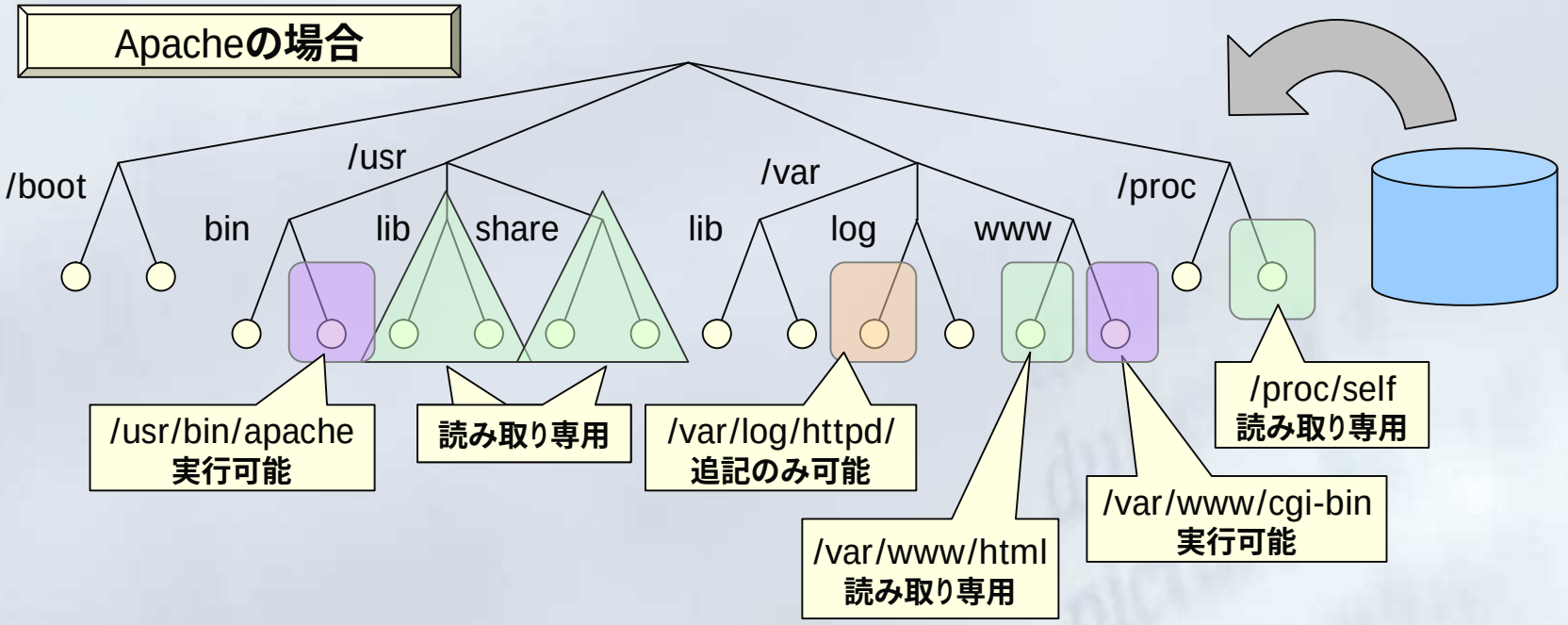
- プログラムごとに、利用可能な資源を絞り込む
  - 保有する"情報資産"を少なくする。
  - クラッカーの"攻撃手段"を少なくする。



※　明示的に許可した資源だけをアクセス可能（ホワイトリスト方式）

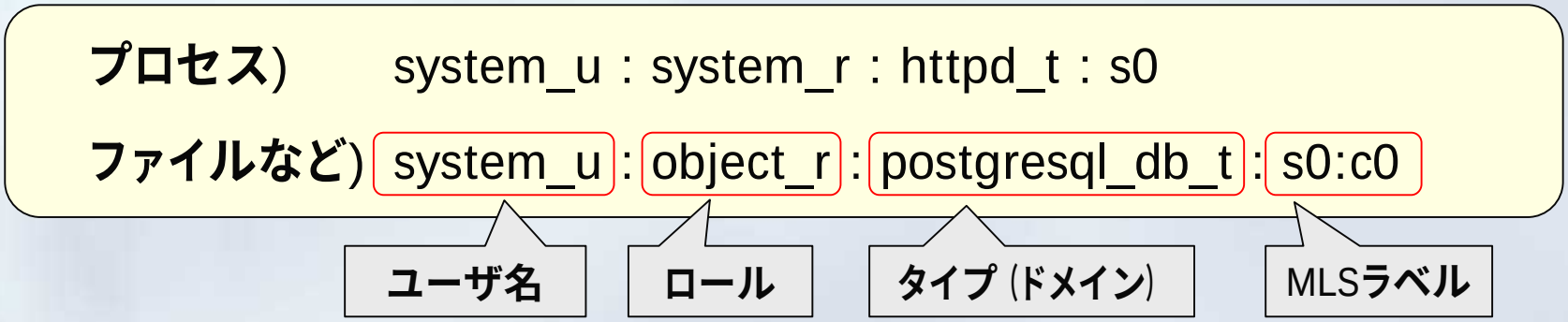
Type Enforcement

- プログラムごとに、利用可能な資源を絞り込む
  - 保有する"情報資産"を少なくする。
  - クラッカーの"攻撃手段"を少なくする。



※　明示的に許可した資源だけをアクセス可能（ホワイトリスト方式）

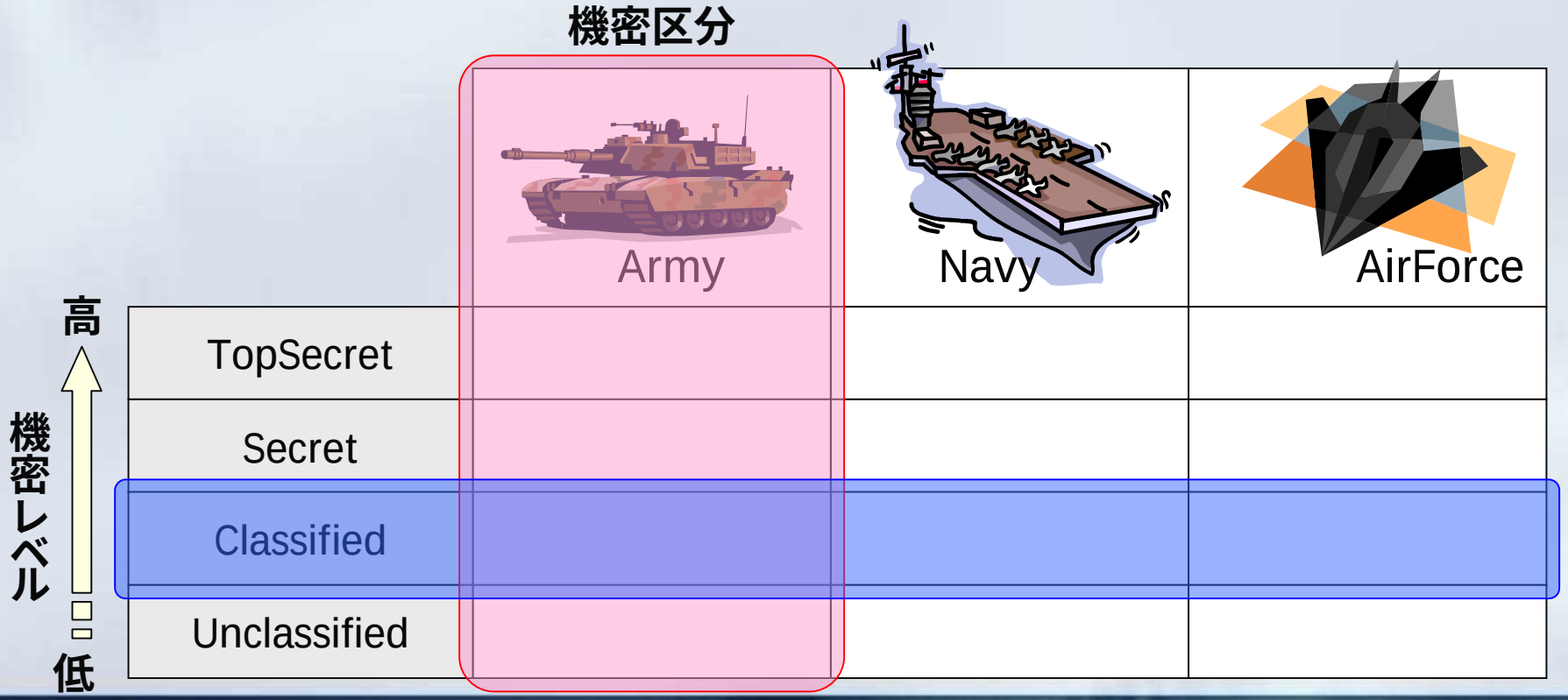
セキュリティコンテキスト



- タイプ／ドメイン
  - プロセスの場合には「ドメイン」と呼びます
  - あるドメインから見て「読み取り専用」「追記のみ可能」etc...を示す識別子
  - この辺の対応関係は、全てセキュリティポリシーで記述されている
- MLSラベル
  - 機密レベルと機密区分を示す
  - 軍用システムに由来する、最も伝統的な”強制アクセス制御”

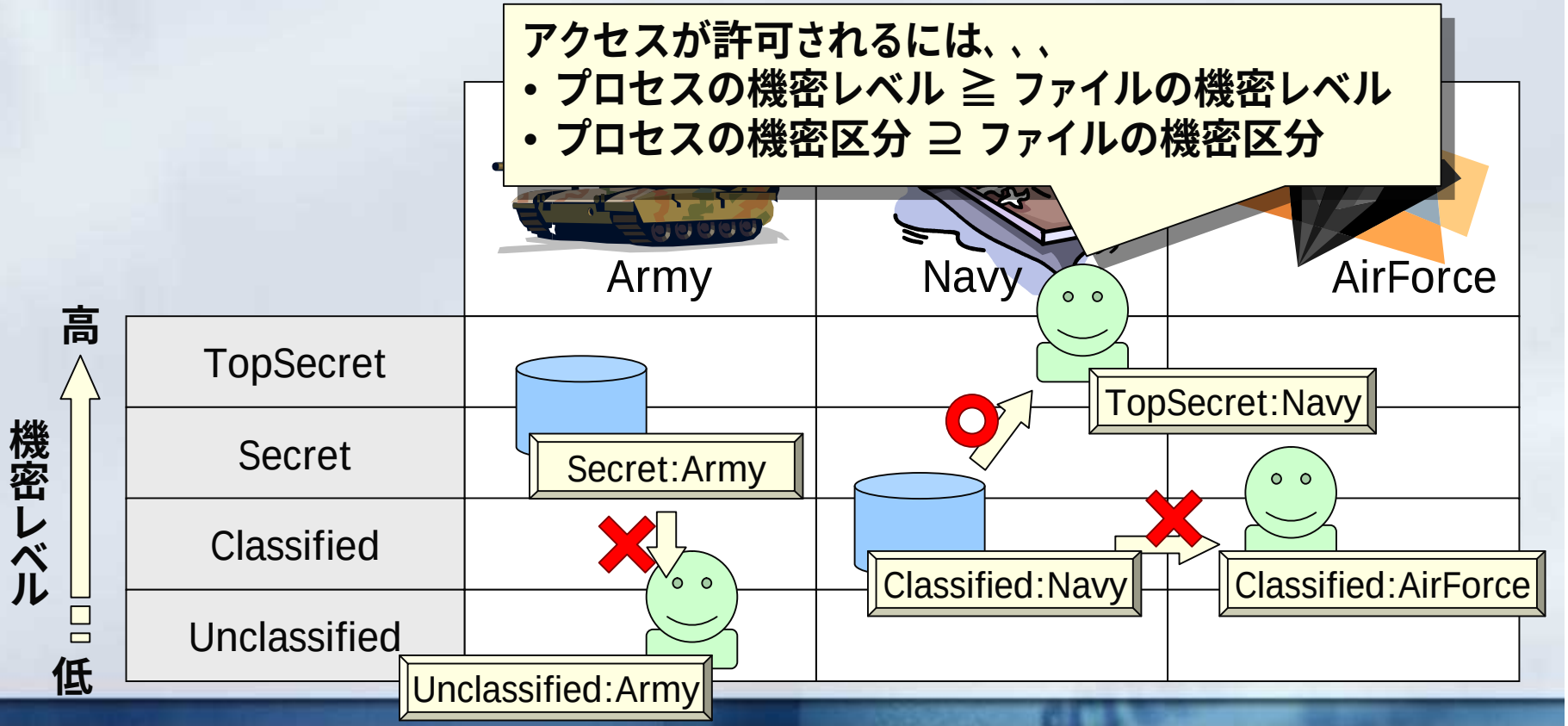
Multi Level/Category Security

- "情報の流れ"を制御する
  - 『機密レベル』の高い情報を、低い所へ流さない
  - 『機密区分』の壁を越えて情報を流さない



Multi Level/Category Security

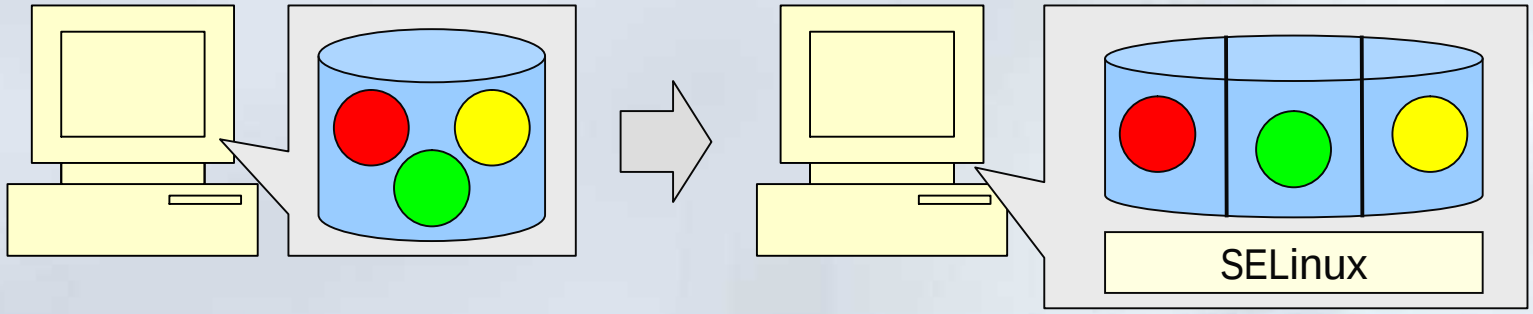
- "情報の流れ"を制御する
  - 『機密レベル』の高い情報を、低い所へ流さない
  - 『機密区分』の壁を越えて情報を流さない



結局、どういう事？

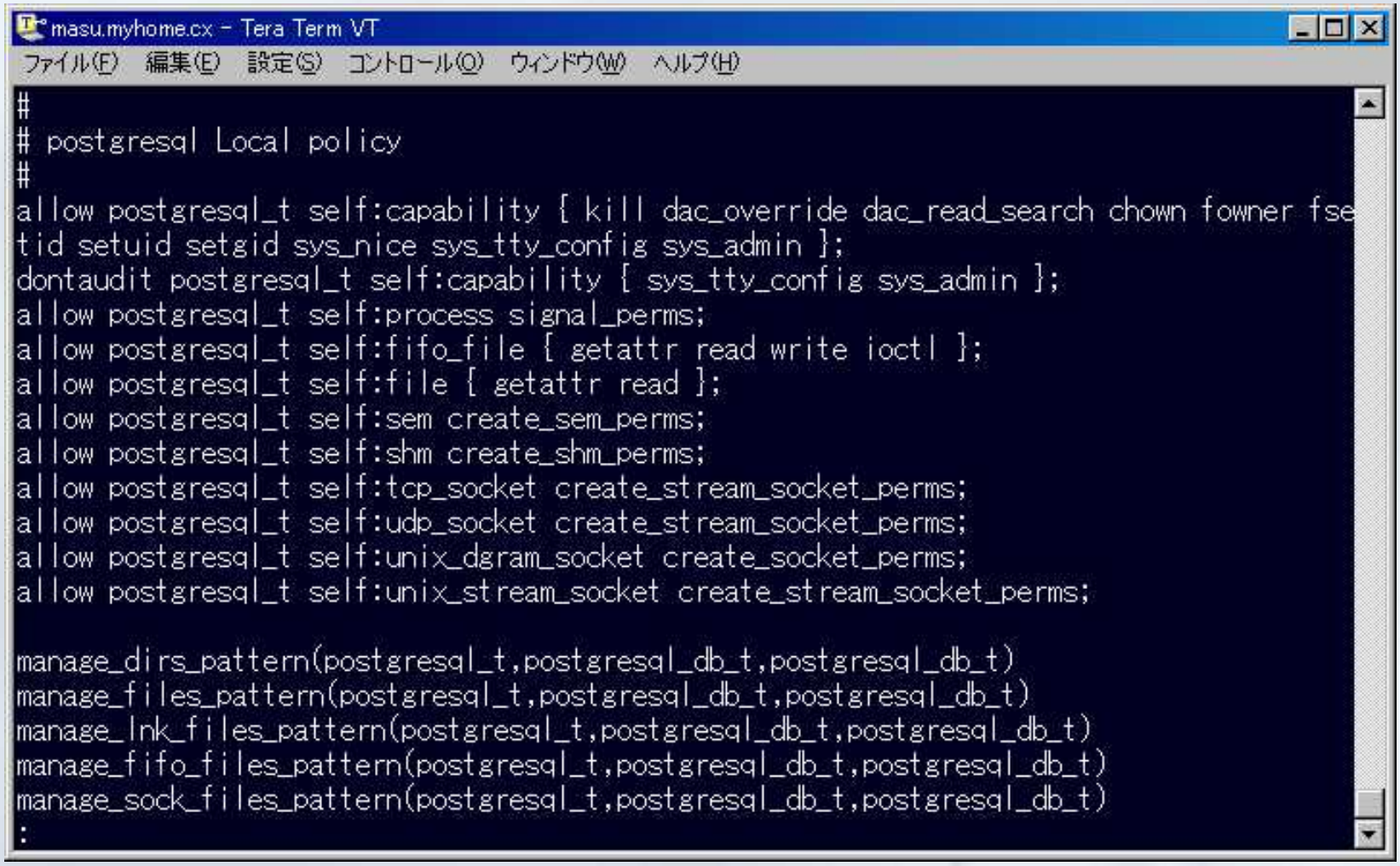
諺：全ての卵を同じ籠に入れるな

- SELinuxは**強制アクセス制御**を提供する
  - TEやMLS/MCSによって、"情報資産"を論理的な壁の中に閉じ込める
    - 仮に一箇所が突破されたとしても、被害を局所化できる
    - 利用可能な資源を限定する事で、攻撃自体が難しくなる
  - "論理的な壁"の一貫性
    - リファレンスモニタとセキュリティポリシーのモデル



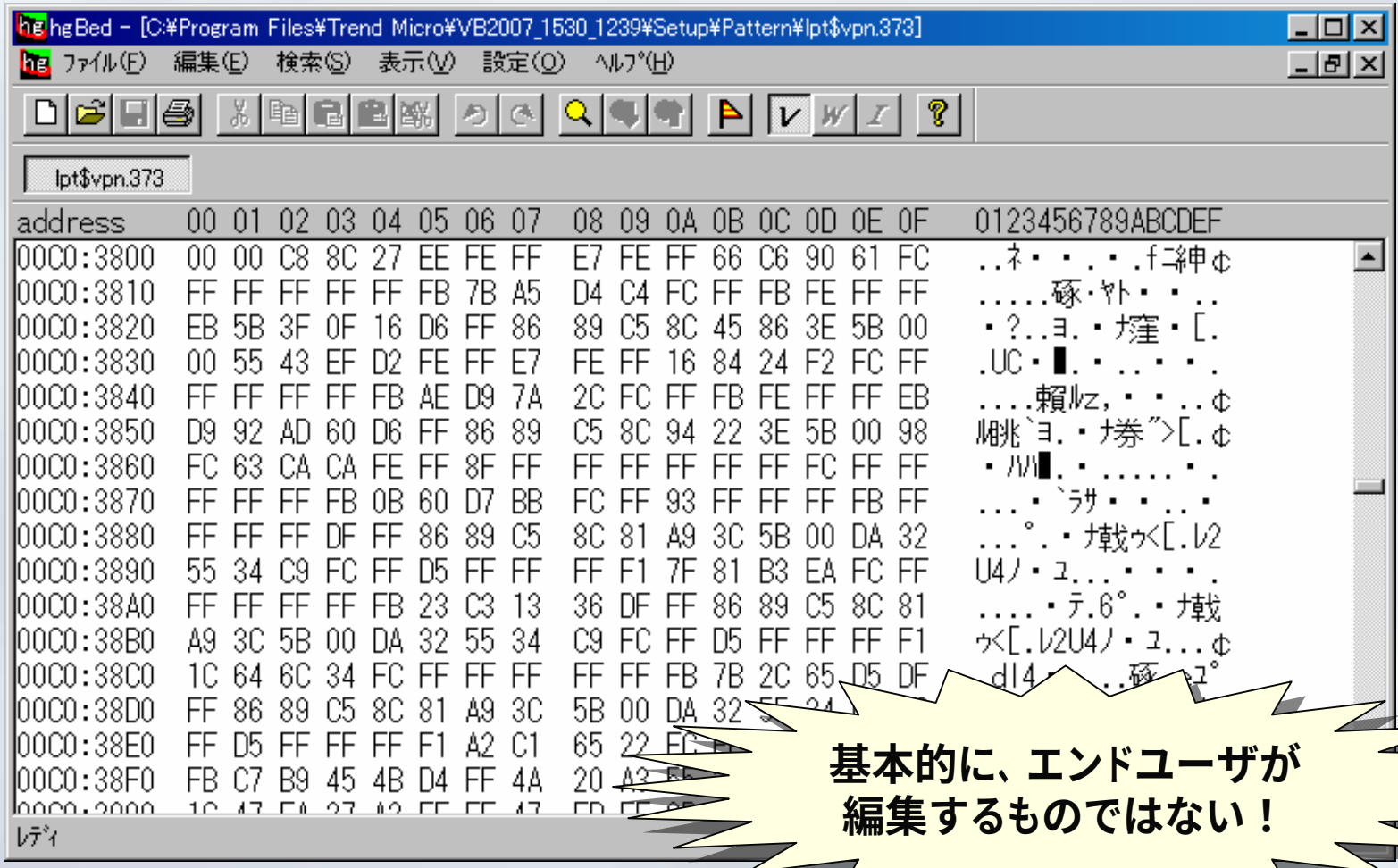
で、SELinuxは ”アリ” なのか？

ポリシーの作成が難しい…？



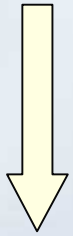
ポリシーの作成が難しい…？

- ↓某社アンチウイルスソフトのパターンファイル



SELinuxポリシー体系

- **3種類の異なるポリシー**
  - Targeted Policy
  - Strict Policy
  - MLS Policy



ぬるい

厳しい

- Targeted Policy
  - FedoraやRedHatEL、CentOSでの標準ポリシー
  - ユーザのシェルなどは今まで通り
    - 全部許可する = unconfinedドメイン
  - 特定のサーバプロセスだけを保護
    - 対象を絞っている = Targeted

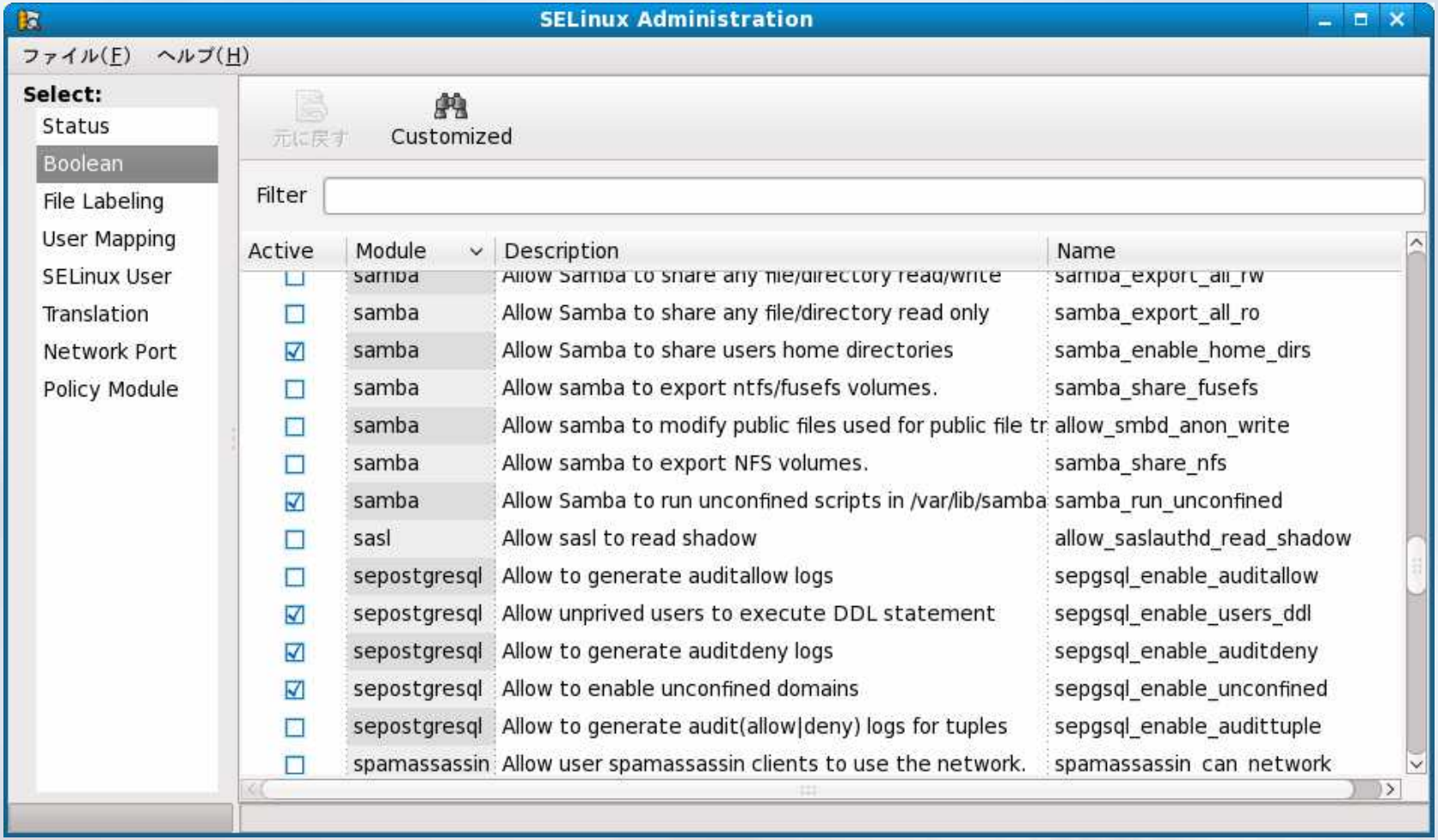
SELinuxの"カスタマイズ"

- booleanの変更
  - boolen
    =セキュリティポリシーの一部を有効化/無効化するためのスイッチ
  - 意味のあるまとまりを単位としてポリシーを修正できる
- ファイルの”タイプ”を変更する
  - 標準のインストールパスを変更
  - 『読み込み専用』→『読み書き可能』へ属性を変更
- MCSを利用する
  - 独自の”機密区分”を設定する事ができる
  - 人に読みやすい"別名"を付ける事ができる

- system-config-selinuxで設定できるんです。

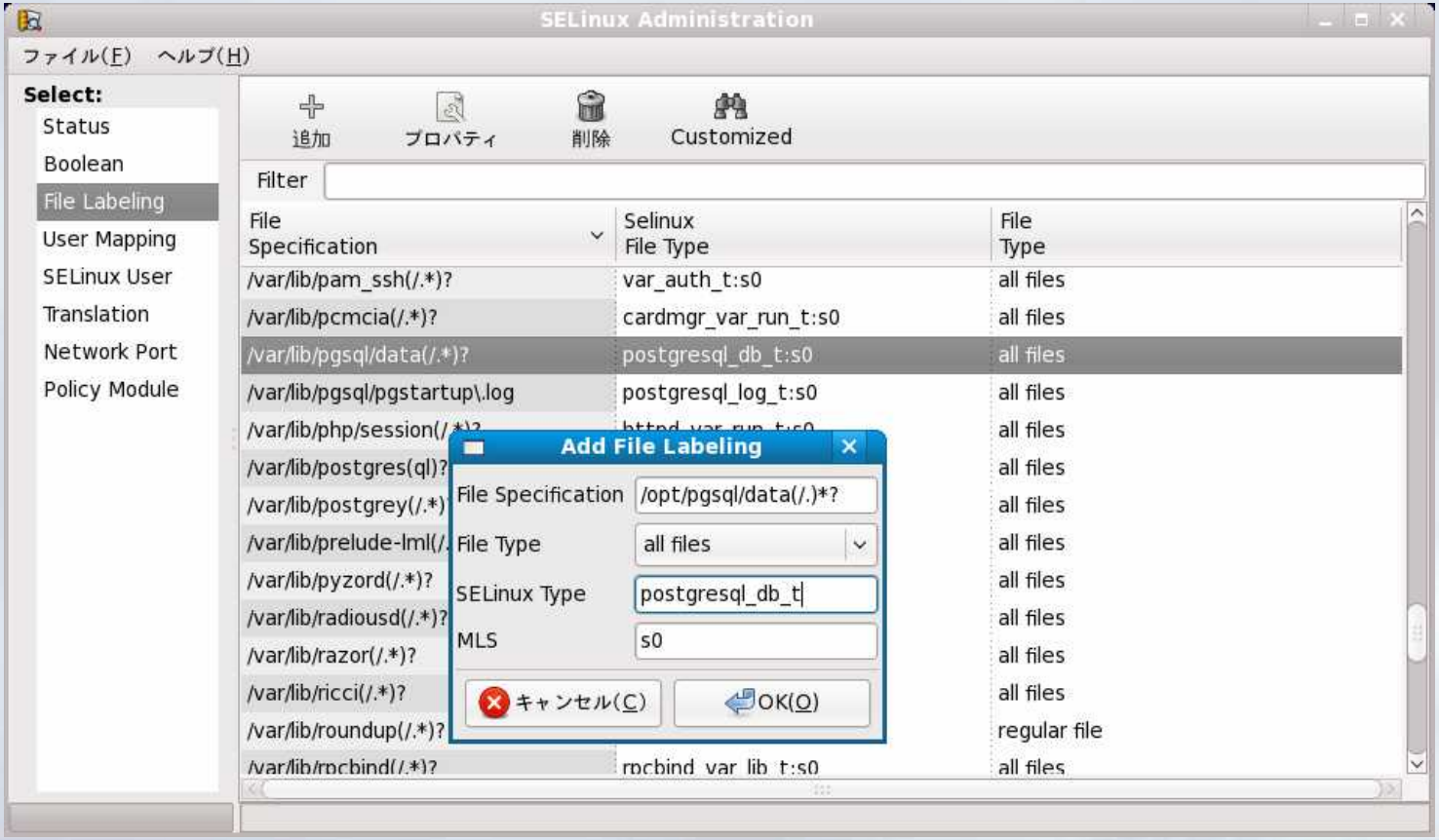
system-config-selinux

- GUIでbooleanのカスタマイズが可能



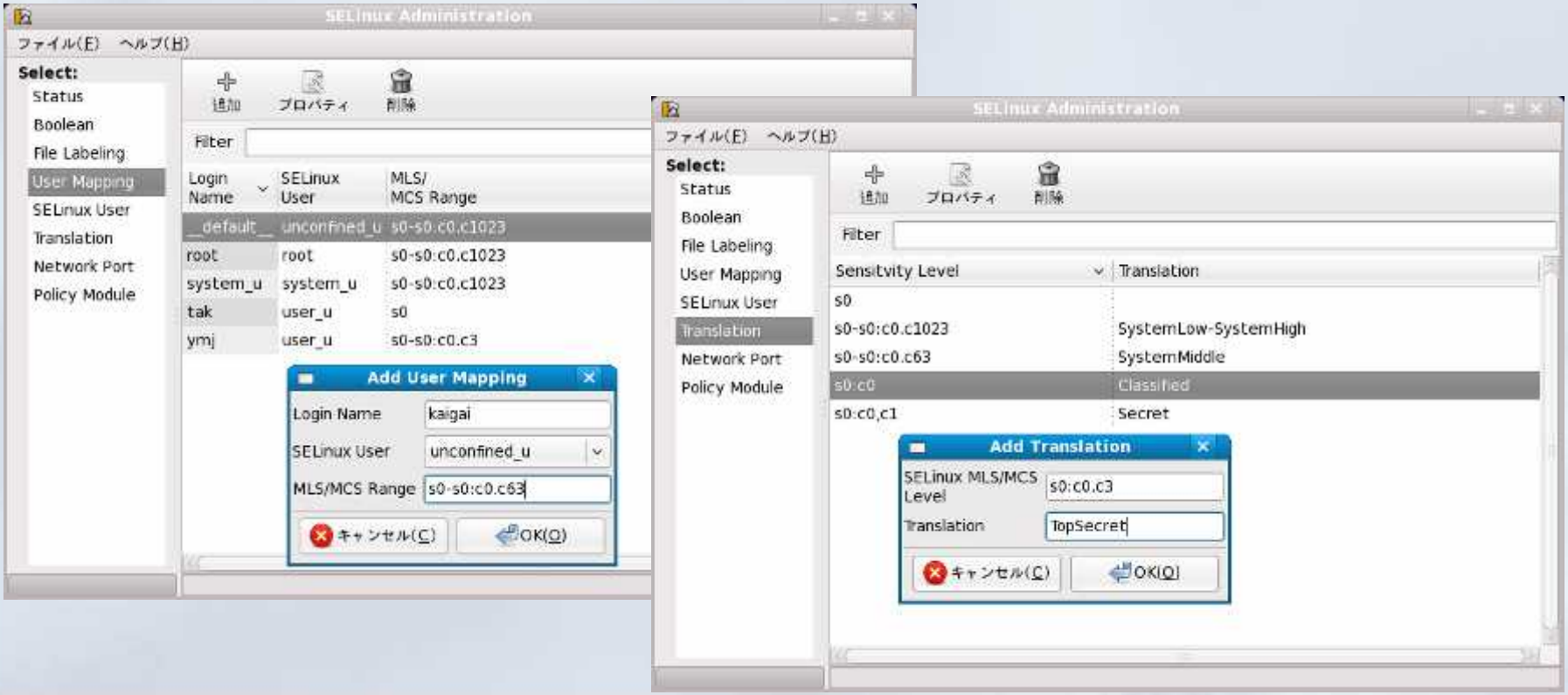
system-config-selinux

- ファイル/ディレクトリの"タイプ"を変更することができる
  - 標準のインストールパスを変更する
  - 『読み込み専用』→『読み書き可能』なタイプに変更する



system-config-selinux

- 各ユーザの"機密区分"を指定できる
- "機密区分"に、読みやすい名前を付けることも可能
  - 『s0:c0.c3』なら『TopSecret』という具合



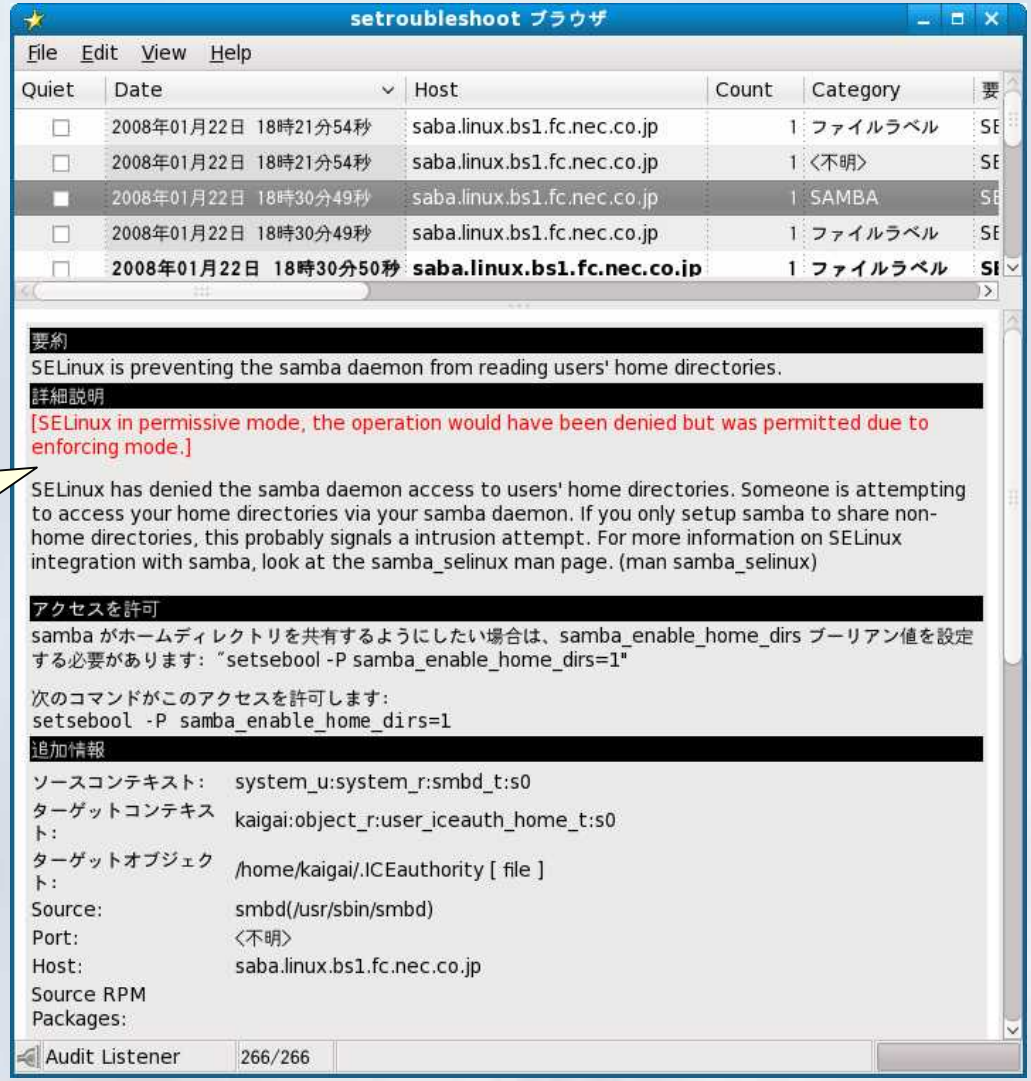
SELinuxでトラブルに遭ったら？

- setroubleshoot
  - ログを解析して、トラブル解決のためのヒントを提示
- **ポリシーモジュールの自動生成**
  - **“最終手段”、ポリシーの修正。**
- **詳しい人に聞いてみよう**
  - **ユーザコミュニティのご紹介**

setroubleshoot

既知の問題点であれば、DBの中から予想される対処方法を検索し、ユーザに提示する。

何かSELinuxが"アクセス拒否"をした場合、ポップアップで通知



ポリシーモジュールの自動生成

(警告) これは最終手段です

cat /var/log/audit/audit.log | **audit2allow** -m hoge -o hoge.te

Auditログから、SELinuxアクセス拒否を抽出
⇒ 拒否されたパターンを"全て許可"するポリシーを自動生成

make –f **/usr/share/selinux/devel/Makefile** hoge.pp

生成されたポリシーを、モジュール形式でビルド
(*) selinux-policy-develパッケージが必要

semodule -i hoge.pp

Policy Packageをインストール

一番手っ取り早い方法

- 詳しい人に聞く。
- selinux-users@selinux.gr.jp には、
  詳しい人が沢山居ます。

- 質問するときのヒント
  - よく分からないけど動かない！→ ×
  - こんな情報を付けてくれると嬉しいです。
    - ディストリビューションの種類
    - /var/log/audit/audit.log の内容
    - selinux-policy のバージョン
    - sestatus –a、getsebool –a、semodule -l の出力結果

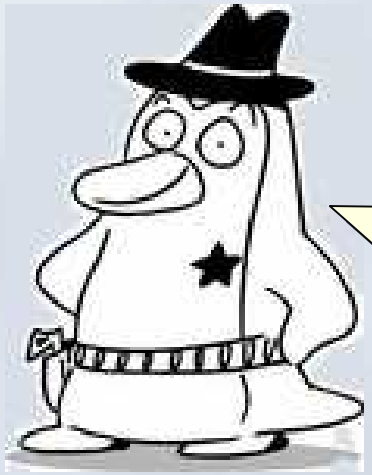
まとめ

- セキュリティの考え方
  - “脆弱性” と ”情報資産”
  - SELinuxは ”アクセス制御” を強化するための機能
  - OSの中に壁を作ることで、
    - 区画（ドメイン）の持つ”情報資産”を減らします
    - 攻撃に使える道具を縛ることで”脆弱性”を減らします
  - そのためのモデルが、TEやMLS/MCS、RBAC
- SELinuxってどうよ？
  - 発想を転換しよう
    - ポリシーを書くのは難しい？ ⇒ 自分で書くから難しい
  - カスタマイズ　　... system-config-selinux
  - トラブルの手助け ... setroubleshoot、audit2allow
  - 最後に頼りになるのは？
    - 日本語で質問できるユーザコミュニティ

業務連絡

- セキュアOS車座集会
  - 於：2F 日本セキュアOSユーザ会ブース
  - セキュアOSを肴に、ワイワイガヤガヤ…。

- 引き続き、この教室で下記セミナーが開催されます



14:00〜14:45

TOMOYO Linuxのある暮らし

〜Linux の勉強からセキュリティ強化まで〜

講師：武田健太郎