TOMOYO® Linux:
あなたのLinuxシステムを理解し護るための実践的な方法

Nov. 29, 2007

原田 季栄
<haradats@nttdata.co.jp>
技術開発本部
株式会社NTTデータ

概　要

- Linuxのセキュリティを振り返る
- TOMOYO Linuxとは
- TOMOYO Linuxと他のLinuxとの比較

事件は起こるもの

- なぜ発生するのかというと:
  1. シェルコード … *が実行されてしまうのは*
  2. バッファオーバーフロー攻撃 … *が可能であるからであり、そこには*
  3. 脆弱性が … *存在するが、その発端は*
  4. 人的エラー … *お手上げ* (これ以上は無理)
- だから、事件が起こらないようにするのは不可能

人間にできることと言えば

- ダメージを軽減すること
- どうやって？
  - 強制アクセス制御（Mandatory Access Control）という発明
  - Linuxを含め多くのオープンソース名OSでも利用可能になってきている
- でも、問題が残ってる…
  - ポリシーの管理は簡単ではない

どうしてポリシーの管理は面倒なの？

- なぜなら
  - 人間側から見るのではなく、最下層であるカーネル側から見なければいけないから
  - 通常はライブラリやミドルウェアによって隠蔽されている詳細についてプログラム開発者側が意識しなければいけないから
  - Linuxのカーネルが理解する流儀と人間が理解する流儀は異なるから
  - 人間もLinuxシステムも強制アクセス制御のためのポリシーを持たずとも存在できるから

唯一の目標への二通りのアプローチ

- 目標
  - 適切なポリシーを入手すること
- アプローチ
  - 「出前」対「自炊」
  - 「出前」とは
    - 誰かが料理して届けてくれる。ユーザ（あなた）はそれを食べるだけ。
  - 「自炊」とは
    - 自分で料理して自分で食べる
  - 言い換えると
    - 「玄人」対「素人」

登場人物を紹介します

- 「玄人」チーム
  - NSA（米国家安全保障局）によるSELinux
  - ユーザはプロが作ったポリシーを使うだけ
- 「素人」チーム
  - TOMOYO Linux
  - ユーザ自身が「自動学習機能」を使ってポリシーを作る
- その中間
  - AppArmor (以前はSubDomainと呼ばれていた)
- 前途有望なルーキー
  - Smack (Simplified Mandatory Access Control Kernel)

ちょっと比べてみたい人は

- http://tomoyo.sourceforge.jp/wiki-e/?WhatIs#comparison (役に立つリンク付き／現在も更新中)

	SELinux	Smack	AppArmor	TOMOYO Linux
Label/Pathname	label		pathname	
Mainline Status	already	#1(Jul 14, 2007) v2 v3 v4 v5 v6 v7 v8 v9 v10 v11 (Nov 8, 2007) now in -mm tree!	#1(Apr 19, 2006) #2 #3 #4(Oct 26, 2007)	#1(Jun 13, 2007) #2 #3 #4 #5(Nov. 17, 2007)
Overview				
Overview	implementation of the research project and architecture, Flask	fairly new attempt towards usable MAC for Linux	Novell had bought the company formerly known as Immunix and ported the technology to SUSE as AppArmor. open source version is also available	developed solely by NTT DATA and was open sourced in 2005
Developed by	NSA	Casey Schaufler	Novell (was)	NTT DATA CORPORATION
Supported by	(mainlined)	project	Mercenary (will be)	project
ISO image for Live CD	N/A	N/A	N/A	w/ Ubuntu 7.10

何が重要なの？

- 私見ですが
  - 「プロフェッショナルによるセキュリティ」という考え方が好きかどうか
  - 自分でやりたいと思う心（あるいはあなたのLinuxシステムへの愛）
  - 管理しなければならないLinuxシステムの数
  - 必要とする機能 (これは簡単)
    - もし、あれもこれも必要ならSELinuxがベストでしょう
- でも、判断する前に、ポリシーを「読んで」ね☺
  - *もし、ポリシーが嫌いだったり内容を理解したくないのなら、やめておくべきです。セキュアOSを使うということは、ポリシーを管理するということなのですから。*

「プロが作ったポリシー」

- 延々と続くLKMLでのAppArmorに関する議論より
- SELinuxの専門家であるKyle Moffet氏曰く
  - *平均的なユーザはセキュリティポリシーを書くべきではない。正直言って、平均的なシステム管理者でさえもセキュリティポリシーを書くべきではない。Webコンテンツへのアクセスを追加する程度の既存のポリシーに対する軽微な修正なら構わないが、それでもプロのシステム管理者／開発者やセキュリティのプロだけがセキュリティポリシーを書くべきである。セキュリティポリシーをぶち壊すのはとても簡単なんだ。*
    - http://lkml.org/lkml/fancy/2007/5/28/359
- SELinuxが存在することは誇れることですが、もし今日使いたいならば「根気」が必要です。その根気があれば、SELinuxはあなたのLinuxシステムへの第一候補となるでしょう。

概　要

- Linuxのセキュリティを振り返る
- TOMOYO Linuxとは
- TOMOYO Linuxと他のLinuxとの比較

開発の動機

- 疑問
  - 誰が自分のLinuxシステムのことを最もよく知っている？
  - 誰が自分のLinuxシステムに責任を持っている？
- 私が思うに
  - それは「あなた」でしょ？
- あなたはセキュリティのプロではないかもしれないし、SELinuxの伝道師でもないかもしれない。でも、あなたのLinuxシステムに関しては専門家になれるはず。
- だから、我々は自分で何とかするためのツールである TOMOYO Linuxを開発しています。

本筋に戻ります

- このプレゼンテーションのタイトルは“TOMOYO Linux:あなたのLinuxシステムを理解し護るための実践的な方法”です。
- 何を護る？何から護る？
  - 悪意ある攻撃から
  - 操作ミスから
  - あなたが秘密にしたい情報を盗み見する奥さんから

目標の設定

- “護る”のは良いとして、何故 “理解”が先にあるの？
- 理解しなければ護ることはできないからね。

目標の設定

- … 私のLinuxシステムの何について理解すればいいの？カーネル2.6.23でUbuntu 7.10を動かしていることを知ってるけど、それでは駄目？
- 駄目です。
- 例えばどんなことを知っていれば良い？
- *gnome-terminal がどのように起動され、どのような処理をするか言えますか？*

目標の設定

- “そんなことに興味ないね。*どうしてそんなことを知っていないといけないの？*”（お静かに願います…）
- あなたのLinuxシステムにどのようなアクセスが必要なのかを伝える必要があります。それが、セキュリティポリシーが機能するために必要なのです。
  - 不便なことですが事実なのです。あなたが護るべきものが何であるかを理解しない限り、あなたはそれらを護ることができません。（プロが考えたセキュリティモデルだけがあっても駄目なのです。）

目標の設定

- きっと、こう言うでしょう
  - “私のLinuxシステムは護りたい。でも、そのために分析してポリシーを書く時間を費やしたくない。”

- おめでとう！
- *TOMOYO Linuxがありますよ。*

では始めます

- どのようにgnome-terminalが起動されたか
- gnome-terminalはどんな処理をしたか

- TOMOYO Linuxであれば
  - それを観ることができます

- デモを始めます

gnome-terminalはどのように起動される？

```
<kernel>
  /sbin/init
    /bin/sh
      /etc/init.d/rc
        /etc/init.d/gdm
          /sbin/start-stop-daemon
            /usr/sbin/gdm
              /etc/gdm/Xsession
                /usr/bin/ssh-agent
                  /usr/bin/x-session-manager
                    /usr/bin/gnome-panel
                      /usr/bin/gnome-terminal
```

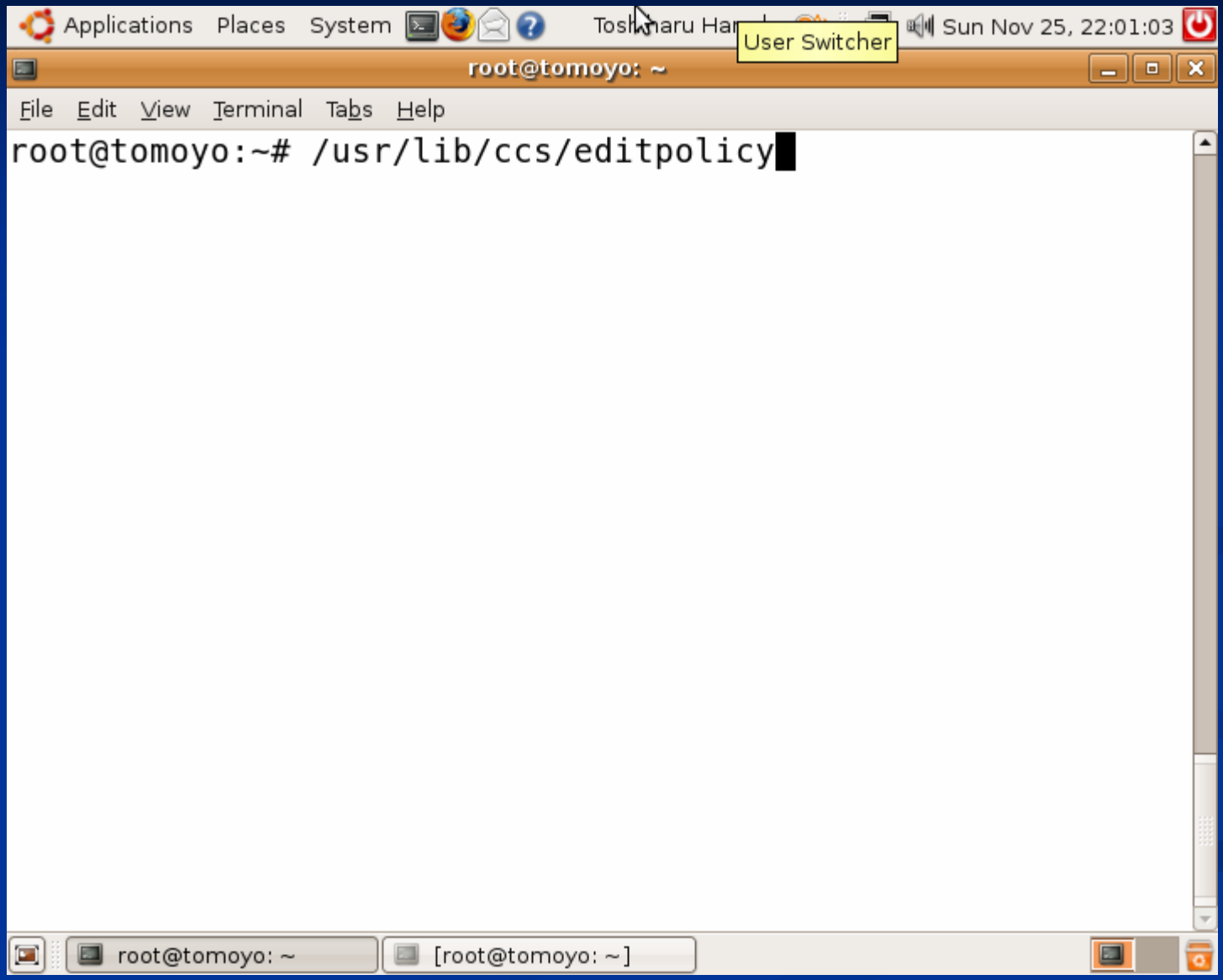
この gnome-terminalは何をしている？

```
exec /bin/bash
exec /usr/lib/libvte9/gnome-pty-helper
read /dev/null
read /dev/urandom
read /etc/fonts/*
read /etc/gnome-vfs-2.0/modules/*
read /etc/nsswitch.conf
read /etc/passwd
read /etc/sound/events/gtk-events-2.soundlist
read /home/toshiharu/.config/user-dirs.dirs
read /home/toshiharu/.gtk-bookmarks
read /home/toshiharu/.ICEauthority
read /home/toshiharu/.Xauthority
read /tmp/gconfd-toshiharu/lock/ior
read /tmp/orbit-toshiharu/bonobo-activation-server-ior
read /usr/lib/gconv/*
read /usr/lib/gnome-vfs-2.0/modules/libfile.so
read /usr/lib/gtk-2.0/*
read /usr/lib/libglade/*
read /usr/lib/pango/1.6.0/module-files.d/libpango1.0-0.modules
read /usr/lib/pango/1.6.0/modules/pango-basic-fc.so
read /usr/share/fonts/*
read /usr/share/gnome-terminal/glade/gnome-terminal.glade2
read /usr/share/icons/*
read /usr/share/mime/*
read /usr/share/pixmaps/gnome-terminal.png
read /usr/share/themes/*
read /usr/share/vte/termcap/xterm
read /usr/share/X11/locale/*
read /var/cache/fontconfig/*
read&write /dev/pts/\$
read&write /tmp/orbit-toshiharu/bonobo-activation-register.lock
```

どうやって知った？

- ポリシーエディタの表示をコピー＆ペーストしただけだよ
- TOMOYO Linuxのポリシーエディタは
  - ドメイン遷移を機構造で表示してくれるよ
  - 各ドメインで発生したアクセス要求を表示してくれるよ
- 観たいでしょ？

どうやって知った？



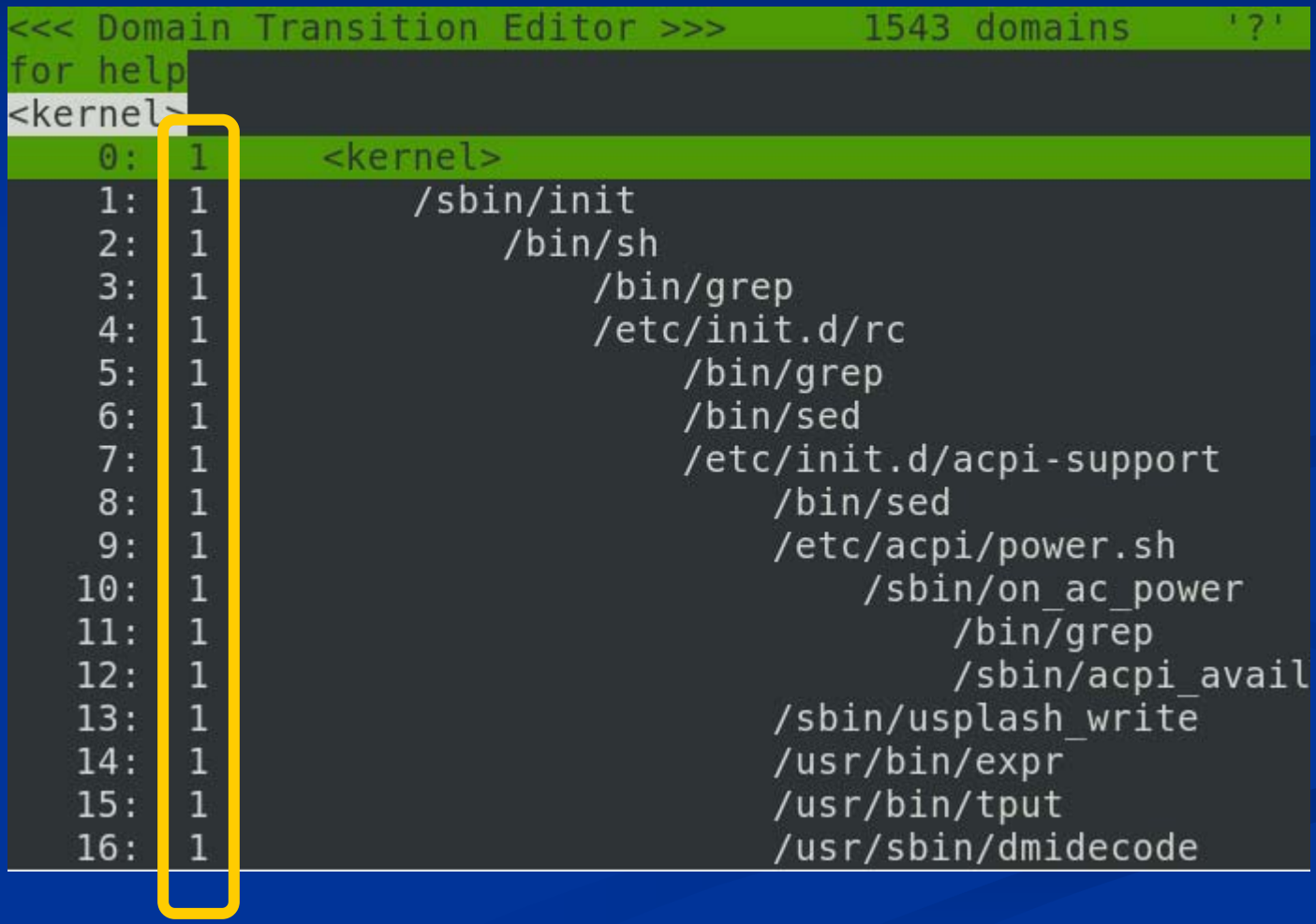
それで？

- TOMOYO Linuxなら準備も根気も要らないよ
  - プロセスがどのように生成されて何をするか（どのような資源にアクセスするか）を教えてくれる世
  - プログラム名ではなくプログラムの実行履歴を用いることでプロセスを詳細に識別できるよ
  - 適切な呼び出し履歴を知ることで、不適切なアクセス要求を検出して阻止することができるよ
- それがこのプレゼンテーションのタイトルである“理解し護る”という意味

１！２！３！　はい、できあがり！

- ポリシーエディタを実行しましょう
  - 1) 保護したいドメインを選択
  - 2)そのドメインのモードを切り替えるために“s”キーを押す
  - 3) プロファイル番号を入力
- プロファイル番号とは
  - /etc/ccs/profile.conf (テキストファイル)
  - 使いたい機能だけを選択してね

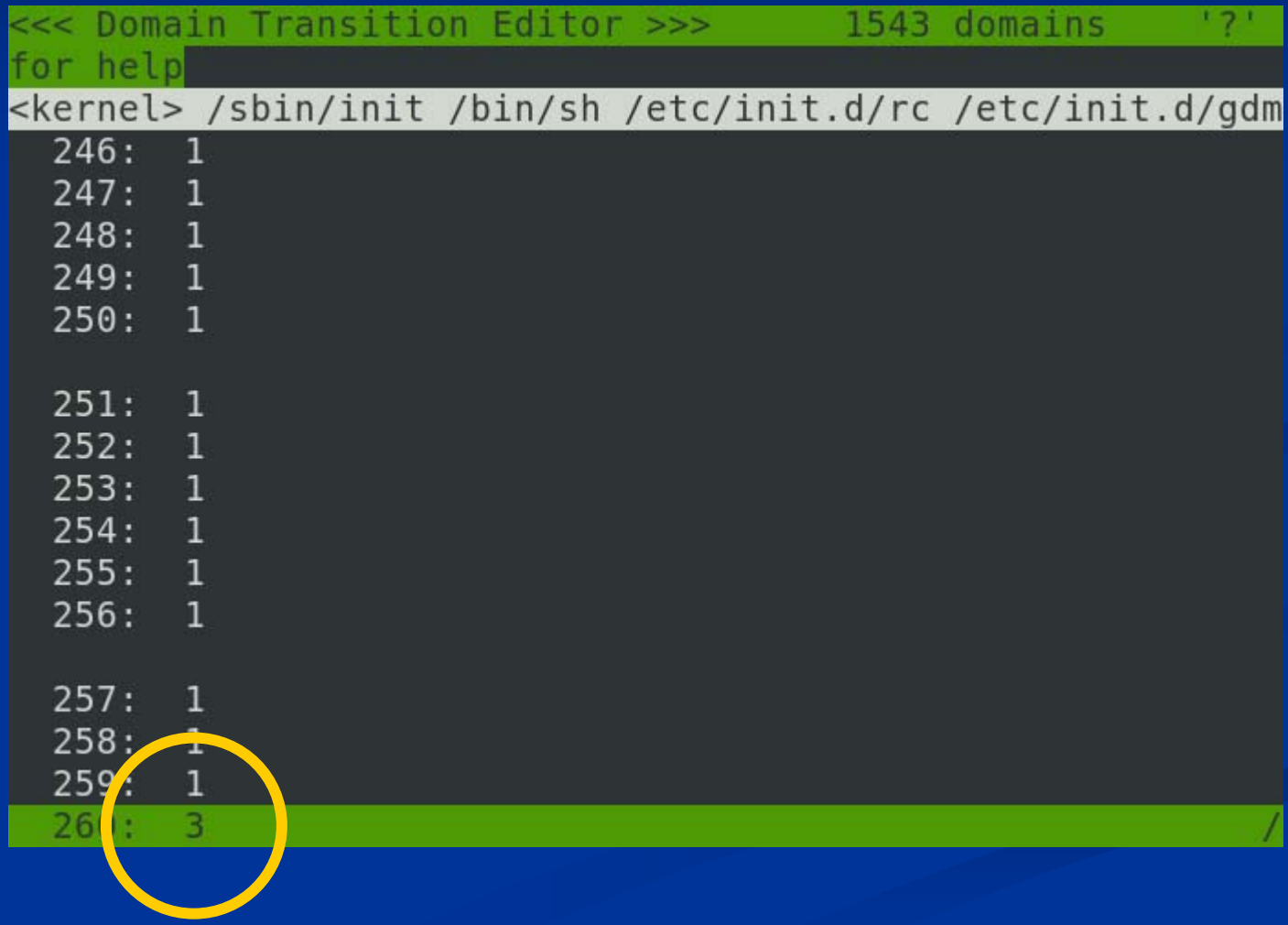
プロファイル番号はどこ？



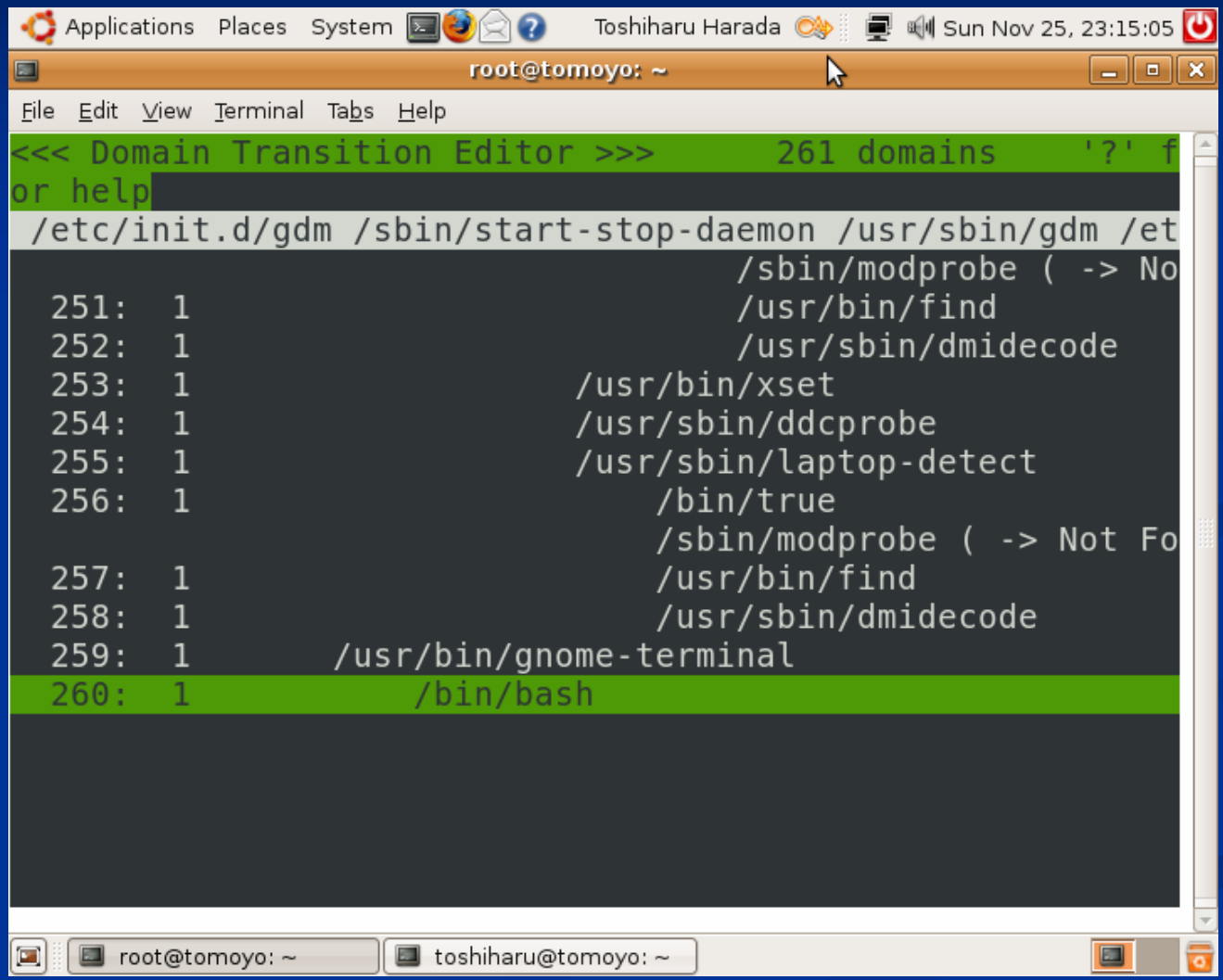
シェルを制限してみよう

```
<<< Domain Transition Editor >>>      1543 domains    '?'
for help
er /usr/bin/gnome-panel /usr/bin/gnome-terminal /bin/bash
  246:  1
  247:  1
  248:  1
  249:  1
  250:  1

  251:  1
  252:  1
  253:  1
  254:  1
  255:  1
  256:  1

  257:  1
  258:  1
  259:  1
Enter profile number> 3
```

シェルを制限してみよう



もう一度観たい？



概　要

- Linuxのセキュリティを振り返る
- TOMOYO Linuxとは
- TOMOYO Linuxと他のLinuxとの比較

SELinuxとの比較

- SELinuxの概要
  - メインラインに取り込まれているセキュリティ強化の実装
  - MLS/MCS/RBACを含むきめこまやかで柔軟なMAC
  - 「セキュリティのことはプロによって設計されたものを使うべし」という考え方→ リファレンスポリシー
  - 魔術師によって設計されサポートされている

SELinuxとの比較

- もし、以下の問題が解決されたらLinuxユーザにとって理想的な解決策
  - リファレンスポリシーが完成する
  - 管理者が適切なラベルを維持することの負担から解放される
- ドメイン単位の制御モード切替が無い（強制／許容の切り替えはシステム全体でしか行えない）

AppArmorとの比較

- AppArmorの概要
  - 昔はSubDomainと呼ばれていた
  - TOMOYOと同様にパス名ベースのMAC (だから我々は兄弟みたいな存在)
  - ドメインはプログラム単位のフラットな構造（TOMOYOでは履歴を用いた木構造）
  - 特定のプログラムだけを保護するように設計されており、システム全体を保護することは想定していない

SELinux, AppArmor, TOMOYO Linux

- いずれもドメインを単位とするMAC
- ドメインの概念が大きく異なる:
  - SELinux
    - ドメインはポリシーを用いて事前に定義されている
    - 階層構造を持たないためフラットである
  - AppArmor (“profile”)
    - ドメインは「Apache」のようにプログラム単位で作成する
    - ドメインはポリシーを用いて事前に定義されている
    - 階層構造を持たない
  - TOMOYO Linux
    - ドメインはカーネルにより自動的に定義され管理される
    - ドメインとはプロセスの実行履歴（あるいは呼び出しの連鎖）である

TOMOYO Linuxを使うと

- 異なる実行履歴を持つ/bin/sh は全く異なるドメインとして扱われる
- TOMOYO Linuxカーネルが自動的に扱ってくれるので、事前に定義する必要が無い
- ドメイン名はプロセスの実行履歴そのものであり、学習しないでも理解できる

もっと情報が必要？

- http://tomoyo.sourceforge.jp/
- http://tomoyo.sourceforge.jp/wiki/