SCIS 2004 The 2004 Symposium on Cryptography and Information Security
Sendai, Japan, Jan.27-30, 2004
The Institute of Electronics, Information and Communication Engineers

SELinux アクセス制御設定項目の安全な統合方法に関する考察
Notes on Secure Integration of SELinux Access Control Configuration Items

末安 克也* 田端 利宏† 櫻井 幸一†
Katsuya Sueyasu Toshihiro Tabata Kouichi Sakurai

あらまし 米国家安全保障局（NSA）は，強力できめ細かなアクセス制御によりセキュリティを強化したセキュア OS として，Security-Enhanced Linux（SELinux）の開発を行っている．しかし，SELinux を用いた強力なアクセス制御を実施するためには，膨大な量のアクセス制御設定を正確に行わなければならない．そこで，SELinux のアクセス制御設定を補助する設定ツールの開発が行われている．SELinux Policy Editor は，SELinux のアクセス制御設定ツールの一つであり，SELinux のアクセス制御設定項目を一部統合して扱うといった機能を持つ．本論文では，SELinux Policy Editor によるアクセス制御設定項目の統合が SELinux システムのセキュリティに及ぼす影響を評価し，その影響を和らげるような，アクセス制御設定項目の安全な統合方法について考察を行う．

キーワード Security-Enhanced Linux, SELinux Policy Editor, セキュア OS, アクセス制御

1 はじめに

電子メールの送受信や Web ページの閲覧をはじめ，さまざまな用途に計算機ネットワークは利用されている．しかし，計算機ネットワークが拡大するとともに，悪意ある攻撃者によって，計算機がネットワークを介した攻撃を受ける危険性も増大してきている．こういった攻撃の多くは，露見したアプリケーションの脆弱性を利用している．あるアプリケーションにどんな脆弱性が潜在しているかわからないため，攻撃者が未知の脆弱性を利用して計算機に侵入することそのものを防ぐことは困難である．したがって，計算機内のデータの漏洩，改ざん，破壊といった直接的な被害を食い止めることが重要になる．これには強力なアクセス制御が有効であるが，それがバイパスまたは無効化されてしまうと無意味であるため，システムの根幹をなすオペレーティングシステム（OS）レベルで強力なアクセスを実現することが望ましい．

Security-Enhanced Linux（SELinux）は，米国家安全保障局（NSA）が Linux OS をベースに開発しているセキュア OS である [1]．SELinux は，元となる Linux OS をはじめとした多くの OS で採用されている任意アクセス制御（Discretionary Access Control, DAC）の機構に対して，強制アクセス制御（Mandatory Access Control, MAC）の機構を採用することでセキュリティの向上を図っている．しかし，SELinux の強制アクセス制御の設定は，任意アクセス制御の設定に比べて複雑になっている．設定ミスはセキュリティを損ねるため，設定はより簡単であるべきである．また，どれだけ強力なセキュリティを実現できても，設定が複雑であるために利用者から敬遠されてしまえば無意味である．

日立ソフトウェアは，SELinux のアクセス制御設定を簡易化し，セキュリティポリシの定義に要する労力を軽減する目的で設定補助ツール SELinux Policy Editor を開発している [2] [3]．このツールは，設定の簡易化の一環として，アクセス制御設定の項目を一部統合しているという特徴を持つ．アクセス制御設定項目の統合は確かに設定を簡易化するが，一方で SELinux の強制アクセス制御のきめ細かさを損なう恐れがあるため，この簡易化がセキュリティに及ぼす影響を明らかにする必要がある．

我々は，以前に SELinux のファイルアクセス制御に関して，SELinux Policy Editor が SELinux のセキュリティに及ぼす影響を明らかにした [4]．本論文では，ファイルアクセス以外の制御に関して，SELinux Policy Editor が SELinux のセキュリティに及ぼす影響を明らかにし，その影響を和らげるようなアクセス制御設定項目の統合方法について考察を行う．

* 〒 812-8581 福岡市東区箱崎 6-10-1，九州大学工学部電気情報工学科，Department of Electrical Engineering and Computer Science, Fuculty of Engineering, Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan, sueyasu@itslab.csce.kyushu-u.ac.jp
† 九州大学大学院システム情報科学研究院，Faculty of Information Science and Electrical Engineering, Kyuhsu University, {tabata, sakurai}@csce.kyushu-u.ac.jp

関連研究として，Jaeger らは，SELinux のセキュリティポリシの完全性保護の解析を，ツールを用いて自動的に行う手法についての提案を行った [5] ．また，Tresys Technology が SELinux 用のツールの開発を行っている [6] ．Tresys は，セキュリティポリシの設定ツールだけではなく，ポリシを解析したり，ユーザの管理を行うためのツールも開発している．これらのツールは，ポリシ管理者が SELinux のセキュリティポリシを理解し，ユーザの追加などの処理を行う助けとなる．

2 Security-Enhanced Linux

本章では，SELinux が持つ機能や，それを実現する機構 [7] [8] について簡単な説明を行う．

2.1 SELinux の機能

DAC ではスーパユーザが存在し，あらゆるアクセス制御設定を無視することができる．しかし，SELinux が採用している MAC では，たとえスーパユーザであってもアクセス制御の対象になる．また，DAC では，ファイルの所有者がそのファイルに対するアクセス権限を自由に決定することができるが，MAC では，アクセス権限の決定権はあらかじめ決められたユーザにのみ与えられる．これにより，アクセス権限を集中管理することができるため，システム全体にセキュリティポリシを徹底することができる．

また，SELinux では，プロセスごとのアクセス制御（Type Enforcement, TE）が実現されている．TE では，プロセスのアクセス権限を，プロセスの所有者単位ではなくプロセス単位で定義することができる．つまり，各プロセスにそれぞれ必要最小限のアクセス権限のみを与えることができる．また，アクセスの客体に関しても，客体単位でアクセス権限を定義することができる．

さらに，SELinux では，ユーザごとのアクセス制御（Role Based Access Control, RBAC）が実現されている．たとえば，Linux OS では，システム管理における重要な処理の権限は全てスーパユーザ（root）が保持している．このため，ある重要な処理をスーパユーザで行っている際に，操作ミスによって現在の処理とは無関係だがシステムにとっては重要であるようなファイルを破壊してしまう恐れがある．一方，RBAC では，スーパユーザの権限を複数の一般ユーザに分散することができる．したがって，操作ミスにより生じる被害を最小限に抑えることができ，重要なファイルを意図しない破壊から守ることができる．

TE や RBAC により，権限は分散され，各々が余分な権限を保持することがなくなる．さらに，MAC により，その権限を改ざんまたは無視することができないようになっている．したがって，これらの機能は攻撃者の攻撃の幅を狭めることができるといえる．攻撃者があるプロセスの制御を奪ったとしても，そのプロセスが必要最小限の権限しか持っていなければ，攻撃者は対象のシステムに大きな打撃を与えることはできない．

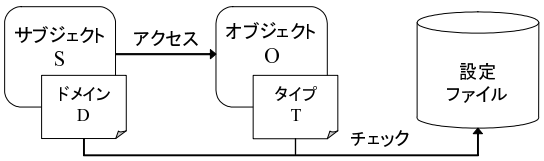

図 1: SELinux で追加されたアクセス制御機構

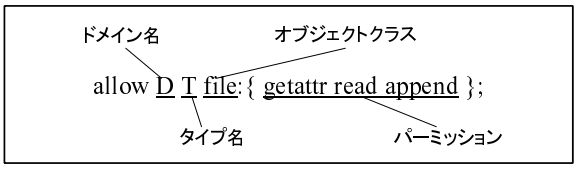

図 2: 権限の記述例

2.2 SELinux の機構

SELinux のアクセス制御機構は，元となる Linux OS のアクセス制御機構の外側に，前項で述べたような機能を持つアクセス制御機構を追加することで実現されている．つまり，両方の機構からアクセス許可を得てはじめて実際にアクセスを行うことができるようになる．本節では，追加された，SELinux 特有のアクセス制御機構について説明を行う．

図 1 に示されるように，SELinux ではアクセスの主体（プロセス）に“ ドメイン ”，アクセスの客体（ファイルなど）に“ タイプ ”というラベルが一つずつ付与されている．アクセス制御の設定は，プロセスやファイルの名前ではなく，それらに付与されているラベルの名前に基づいて行われる．図 2 にはアクセス制御の設定例が示されている．この例の場合，ドメイン D を持つ主体に対して，タイプ T を持つファイルに 3 種類（属性取得，読み取り，追記）のアクセスをする権限が与えられる．ラベルは，“ _t ”で終わる文字列で表現される．複数の主体（客体）に同じラベルが付与された場合，それらが同等のアクセス権限情報を付与されたことを意味する．逆に，全ての主体に異なるドメインを付与することもできるため，プロセスごとのアクセス制御（TE）を実現できることがわかる．

次に，ロールの概念について説明する．ロールは，その名の通り役割を意味し，ユーザと関連付けられる．ロールは，“ _r ”で終わる文字列で表現される．一つのロールを複数のユーザと関連付けることも，一人のユーザに複数のロールを関連付けることも可能である．ユーザはログイン時にロールを選択し，そのロールに対応するド

メインでユーザシェルが起動する．たとえば，あるユーザが user_r というロールでログインした場合，シェルは user_t ドメインが持つ権限に基づいてアクセスを行うことができるようになる．したがって，ユーザに関連付けられるロールごとに異なった権限をユーザシェルに与えることができるので，ユーザごとのアクセス制御（RBAC）が実現できることがわかる．

さらに，プロセスにドメインを付与する手段であるドメイン遷移について説明する．初期化プロセスを除く全てのプロセスは別のプロセス（親プロセス）によって生成される．特に指定がなければ，子プロセスは親プロセスのドメインを受け継ぐが，明示的にドメイン遷移が定義されていると，子プロセスは新たなドメインを持って生成される．これにより，子プロセスに親プロセスと異なる権限を与えることができる．

以上の設定は，アクセス制御設定ファイルを編集することによって行われる．つまり，アクセス権限の決定権を与えたいユーザにのみアクセス制御設定ファイルの書き込みを許すことで，強制アクセス制御（MAC）を実現できることがわかる．また，権限は必ず明示的に定義されていなければならない．つまり，アクセスの許可はそれが設定ファイル内で定義されている場合にのみ行われ，定義されていないアクセスは一切許可されない．

このように，SELinux ではプロセスごと，ユーザごとのきめ細かなアクセス制御を行うことができる．さらに，パーミッションの種類も，Linux に比べて多くなっている．たとえば，Linux OS におけるファイルアクセスのパーミッションは読み，書き，実行の 3 種類であるが，SELinux におけるファイルアクセスのパーミッションは，読み，書き，実行に加えて，ファイルの追記や作成，ファイル属性の取得や設定，ファイル名の変更など合計で 17 種類に及ぶ．しかし，きめ細かな設定にはそれだけ多くの労力を要するので，SELinux ではマクロの定義を許している．マクロはユーザが自分で定義することもできるが，SELinux があらかじめ例として与えているポリシ（example policy）の中でも定義され，使用されている．

3 SELinux Policy Editor

前章で述べたように，SELinux のアクセス制御設定は，セキュリティポリシが記述されているファイルを編集することで行われる．しかし，このファイルはテキストベースであるため，ファイルの内容から現在の設定状況を把握することは困難であり，設定漏れや誤設定によって SELinux の強力なアクセス制御が十分な効果を発揮できなくなる可能性が生じる．そこで，日立ソフトウェアは，SELinux の設定を補助するツールである SELinux Policy Editor の開発を行っている．

SELinux Policy Editor の主な機能について説明する．

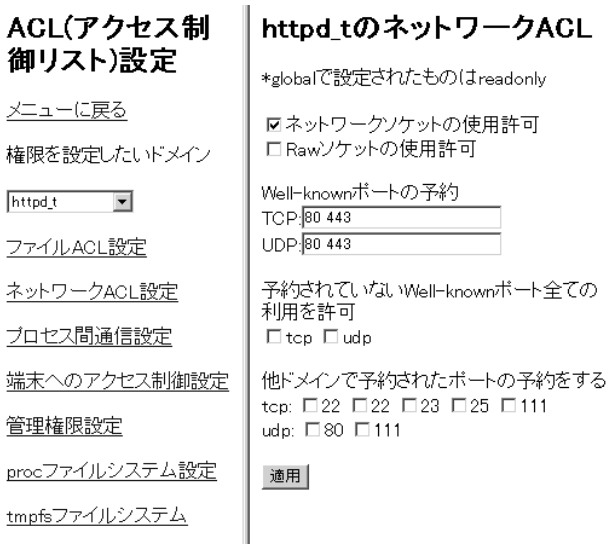

図 3: SELinux Policy Editor の設定画面

まず，Graphical User Interface（GUI）を通した設定が可能であることが挙げられる．これにより，パーミッション付与やドメイン遷移といった現在の設定状況を，視覚的に把握することができる．図 3 は，SELinux Policy Editor の設定画面の例である．権限を設定したいドメインを左のフレームで選択してから，そのドメインに付与する権限を，右のフレーム内にあるチェックボックスなどを利用して設定していく．もう一つの機能として，設定項目を一部統合して扱っていることが挙げられる．たとえば，SELinux におけるファイルアクセスのパーミッションは 17 種類存在するが，SELinux Policy Editor のファイルアクセス制御の設定画面では，パーミッションは 4 種類しか存在しないように見える．SELinux Policy Editor では，17 種類のパーミッションのうち 12 種類を 4 種類に統合したものを利用者に見せ，チェックされるとそれに対応するパーミッションを実際に付与するようになっている．統合前後のパーミッション間の対応関係を表 1 に示す．なお，統合されていない 5 種類のパーミッションは，ラベルを付与する権限などを表わすものであり，他の設定画面を通して定義される．また，ディレクトリアクセスのパーミッションも同様に 4 種類に統合されており，対応関係を表 2 に示す．表 1 や表 2 で列挙されている SELinux のパーミッションがそれぞれどのようなアクセスを許可するものであるかについては，日立ソフトウェアによる SELinux の実装方法調査 [9] の中で解説されている．以上のように，SELinux の設定項目を一部統合することで，SELinux の設定は簡易化される．

4 設定項目の統合による影響

SELinux Policy Editor は，SELinux の設定を簡易化するために一部の設定項目を統合して扱っているため，

表 1: ファイルアクセスパーミッションの統合

統合パーミッション	対応するSELinuxのパーミッション
r	read, getattr, ioctl, lock
w	write, setattr, append, create, unlink, link, rename
x	execute
s	getattr

表 2: ディレクトリアクセスパーミッションの統合

統合パーミッション	対応するSELinuxのパーミッション
r	read, getattr, ioctl, lock
w	write, setattr, append, create, unlink, link, rename, add_name, remove_name, reparent, rmdir
x	execute
s	getattr, search, read

SELinux のアクセス制御設定のきめ細かさが損なわれてしまっている恐れがある．本章では，SELinux Policy Editor を用いて構築されたセキュリティポリシの解析を行い，このツールによる設定項目の統合が SELinux システムのセキュリティに及ぼす影響を明らかにする．なお，SELinux では，強制アクセス制御によって計算機内のデータの漏洩や改ざんといった攻撃に対する防御を高めることはできるが，サービス停止攻撃（DoS 攻撃）に対する防御はあまり考えられていない．したがって，本論文では，システムの動作を妨害するだけの攻撃ではなく，計算機内のデータの漏洩や改ざんといった攻撃が成立するかどうかを SELinux のセキュリティの基準としている．そして，ポリシ解析の対象は，Web サーバソフトウェアである Apache の example policy [10] とした．Apache が世界的に広く利用されているソフトウェアであることに加え，システムの管理者がサーバに SELinux を導入することを考えると，一からセキュリティポリシを構築するよりも，example policy を自分の環境に応じて編集する方がより容易で現実的なためである．

図 3 の左フレームにも示されるように，SELinux Policy Editor では，SELinux の ACL（アクセス制御リスト）の設定を，次のように分類して行っている．

- ファイル ACL 設定
- ネットワーク ACL 設定
- プロセス間通信設定
- 端末へのアクセス制御設定
- 管理権限設定
- proc ファイルシステム設定
- tmpfs ファイルシステム設定

各分類ごとに，まず設定内容を簡潔に説明し，それから SELinux Policy Editor による設定項目の統合内容を示し，最後にその統合が SELinux システムのセキュリティに及ぼす影響について論じる．

(1) ファイル ACL 設定

ファイル ACL 設定では，通常のファイルやディレクトリに対する，読み，書き，実行，探索といった権限を設定することができる．これらは，データの漏洩や改ざんに直結する権限であり，重要である．表 1 や表 2 に示されている統合パーミッション（r, w, x, s）は，ここで用いられているものであり，表 1 は，ファイル ACL 設定に関して SELinux Policy Editor が行っている設定項目統合の内容を示しているといえる．このパーミッションの統合が SELinux システムのセキュリティに及ぼす影響については，我々の以前の研究で明らかになっている [4] ．以下に，結果だけを示す．

- 一部の追記専用ファイルの改ざんおよび，一部の追記・上書き専用ファイルの削除の危険性
- 一部のディレクトリの構成ファイル名の漏洩の危険性

(2) ネットワーク ACL 設定

ネットワーク ACL 設定では，ネットワークソケットの使用許可や，各ポートの使用許可の設定を行うことができる．SELinux Policy Editor では，ソケットの使用許可について統合が行われており，送信のみ（または受信のみ）を許可することはできなくなっている．しかし，ポートに関しては，SELinux 本来の設定と同様に，ポート番号ごとに使用許可を設定することが可能である．Apache の場合は，80 番ポート（HTTP）を通した送受信の権限が必要なので，問題は生じない．他のアプリケーションに関しても，最初から送受信両方の権限が必要である場合が多いと考えられる．

(3) プロセス間通信設定

プロセス間通信設定では，共有メモリやパイプへのアクセス権限，シグナルの送信権限など，プロセス間通信に必要な権限の設定を行うことができる．SELinux Policy Editor では，共有メモリやパイプなどのプロセス間通信に用いる客体へのアクセス権限が統合されている．客体の種類やラベル名は一切統合されていないが，パーミッションの種類が 1 つに統合されているため，ある客体に対して全てのアクセスを許可するか許可しないかのどちらかしか選択できない．Apache の example policy において，プロセス間通信の権限は，デーモンや CGI のプロセスなど，Apache のプロセス間で行われるものしか

定義されておらず，統合の影響が他のアプリケーションに及ぶことはない．

(4) 端末へのアクセス制御設定

端末へのアクセス制御設定では，ローカルまたはリモートでのログイン時に作成・利用される端末または擬似端末に対するアクセス権限を設定できる．SELinux Policy Editor では，端末または擬似端末の生成に必要な権限はそれぞれひとまとめにされており，さらに，各端末の操作に必要な権限は読み出しと書き込みの 2 種類に統合されている．端末の生成権限に関しては，最低限必要な権限がまとめられているだけなので，問題は生じない．端末の操作権限に関しては，読み出しと書き込みの 2 種類が存在していれば致命的な問題には繋がらないと考えられる．

(5) 管理権限設定

管理権限設定では，リブートやトレースに必要なシステムコール，または SELinux 制御用コマンドを実行する権限の設定が行われる．また，全ファイルの読み出しや書き込みといった権限もここで設定することができる．前者に関しては，SELinux Policy Editor による統合は行われていない．後者に関しては，セキュリティを重視するのであれば，そもそもチェックを行うべきではない，全ファイルへの権限を一括して設定するのはセキュリティ上危険であるから，ファイル ACL 設定でファイルごとに権限を設定するべきである．なお，ファイル ACL 設定だけでも全ファイルの読み出し権限を設定することは可能である．

(6) proc ファイルシステム設定

proc ファイルシステムは，起動中のプロセスの情報をはじめとした，カーネルのシステム情報を格納するファイルシステムであり，/proc 以下に存在する．proc ファイルシステム設定では，この proc ファイルシステムに対するアクセス権限を設定することができる．SELinux Policy Editor では，端末へのアクセス制御設定と同様に，アクセス権限は読み出しと書き込みの 2 種類に統合されている．また，アクセスの対象は，自プロセスの情報，他プロセスの情報，システム共通で利用する情報，カーネルログ，その他のシステム情報の 5 種類に統合されている．proc ファイルシステム内には，システムの情報を保持する重要なファイルが多く存在するため，アクセスの種類や対象の統合が SELinux システムのセキュリティを損ねるおそれがある．しかし，ファイル ACL 設定とは違い，設定項目の統合が計算機内のデータの漏洩や改ざんの危険性に直結しているとはいえない．

(7) tmpfs ファイルシステム設定

tmpfs ファイルシステムは，仮想メモリベースのファイルシステムである．tmpfs ファイルシステム設定では，この tmpfs ファイルシステムに対するアクセス権限を設定することができる．SELinux Policy Editor では，端末へのアクセス制御などと同様に，アクセス権限が読み出しと書き込みの 2 種類に統合されている．しかし，Apache の example policy では，SELinux Policy Editor による統合内容と同じ組み合わせでのみパーミッションが付与されているため，問題は生じない．

5 より安全な統合方法の考察

前章では，SELinux Policy Editor によるアクセス制御設定項目の統合が，SELinux システムのセキュリティに与える影響について論じた．本章では，それらの検討結果に基づいて，設定項目のより安全な統合方法について考察を行う．

安全性だけを考えるならば，余分な権限が与えられる原因であるアクセス制御設定項目の統合を一切行わず，GUI のみによる設定補助を行うことが望ましいといえる．しかし，統合を一切行わない場合，GUI を通したとしても設定項目そのものは複雑なままである．したがって，セキュリティポリシの管理者には，設定に要する労力だけではなく，各設定項目に関する十分な知識が求められることになる．小規模なシステムのセキュリティを向上させるために SELinux を導入する場合，管理者が SELinux の設定に関する知識を得て，実際に設定を行うだけの余裕がない可能性がある．そこで，致命的な問題が生じないという条件を満たした上で，SELinux のアクセス制御設定をできるだけ簡易化するような統合方法について考える．

前章で論じたように，SELinux Policy Editor による設定項目統合では，統合により過剰に与えられた権限がセキュリティ上無視できない問題を引き起こす場合がある．そこで，前章で述べられた過剰な権限のうち，セキュリティ上大きな問題となるものを挙げ，それらを統合の対象から外した統合方法を考えることにより，より安全かつ設定の労力を抑えた設定項目統合方法を得る．

計算機内のデータの漏洩や改ざんといった攻撃に直結するのは，ファイル ACL 設定における統合である．一方，ネットワーク ACL 設定やプロセス間通信設定などのファイル ACL 設定以外の設定についても統合は行われており，システムのセキュリティに対する影響も，前章で述べたように存在する．しかし，その影響は，設定に必要な知識および労力を抑える効果と比較して小さく，システムにとって致命的であるとはいえないと考えられる．

ファイル ACL 設定に関して問題となる過剰な権限は，ファイルやディレクトリの上書きおよび生成・削除権限

と，ディレクトリの構成ファイル名の取得権限である．これらの権限を統合対象から外すには，SELinux Policy Editor による統合に対して，ファイルやディレクトリの append 権限，write 権限，create 権限を分離し，さらにディレクトリの search 権限と read 権限を分離すればよい．これらの権限の統合と，SELinux システムに生じるセキュリティ上の問題との関係は，論文 [4] の中で述べている．append，write，create の権限は表 1 の統合パーミッション w で統合されているため，これらの権限を分離するためには，append 権限と write 権限を別の統合パーミッションとして定義すればよい．一方，ディレクトリに対する search，read の権限は表 2 の統合パーミッション s で統合されているため，統合パーミッション s から read 権限を外すことで権限の分離を行うことができると考えられる．

この新たな統合方法では，セキュリティ上致命的な問題こそ生じないが，それはセキュリティ上の問題が一切生じないことを意味するわけではない．管理者に十分な知識と労力が与えられるような大規模なシステムに SELinux を導入する場合には，設定項目を一切統合せずに設定を行うことも十分に可能である．したがって，本章で示した新たな統合方法を用いた大雑把な設定をまず行い，その後任意で詳細な設定を行うこともできるような仕組みが望ましいといえる．最初に，統合された設定項目を用いた大雑把な設定を行うことで必要最小限のセキュリティを実現でき，さらに，管理者の能力やシステムのセキュリティ要件などの条件次第では，詳細な設定を追加することもできる．

6 まとめ

本論文では，SELinux の設定ツールである SELinux Policy Editor による SELinux 設定項目の統合について，それが SELinux システムのセキュリティに及ぼす影響について論じた．そして，設定に要する労力とセキュリティのバランスを考慮した上で，設定項目の新しい統合方法について考察を行った．その結果，管理者の能力やシステムのセキュリティ要件に沿った設定を行うためには，大雑把な設定と詳細な設定を組み合わせることが効果的であるという結論を示した．

この考察は，Apache の example policy に対してアクセス制御設定項目の統合を施したものを解析した結果に基づいて行われている．したがって，本論文で示した大雑把な設定がシステム全体にとって必要最小限なセキュリティを実現できることを，他のポリシ（他のアプリケーションの example policy など）に関して検証する必要があると考えられる．

参考文献

[1] NSA, Security-Enhanced Linux.
URL=http://www.nsa.gov/selinux/

[2] 日立ソフトウェアエンジニアリング株式会社, SELinux Policy Editor.
URL=http://www.selinux.hitachi-sk.co.jp/tool/selpe/selpe-top.html

[3] 中村 雄一, 鮫島 吉喜 “Security-Enhanced Linux のアクセス制御ポリシ設定の簡易化,”2003 年 暗号と情報セキュリティシンポジウム (SCIS2003) 予稿集, Vol.II, pp.831–836, Jan, 2003.

[4] 末安 克也, 田端 利宏, 櫻井 幸一 “簡易化されたポリシに基づいた SELinux アクセス制御の安全性評価,” コンピュータセキュリティシンポジウム 2003(CSS2003) 論文集, pp.253–258, Oct, 2003.

[5] T. Jaeger, R. Sailer, X. Zhang “Analyzing Integrity Protection in the SELinux Example Policy,”Proc. of the 12th USENIX Security Symposium, pp.59–74, Aug, 2003.

[6] Tresys Technology, SELinux research.
URL=http://www.tresys.com/selinux/index.html

[7] P. Loscocco and S. Smalley “Integrating Flexible Support for Security Policies into the Linux Operating System,”Proc. of the FREENIX Track of the 2001 USENIX Annual Technical Conference, pp.29–42, Jun, 2001.

[8] S. Smalley “Configuring the SELinux Policy,”NAI Labs Rep. 02-007, Feb, 2002.
URL=http://www.nsa.gov/selinux/policy2-abs.html

[9] 日立ソフトウェアエンジニアリング株式会社 “オペレーティングシステムのセキュリティ機能拡張の調査,”IPA/ISEC 情報セキュリティ関連の調査・開発に関する公募.
URL=http://www.ipa.go.jp/security/fy13/report/secure_os/secure_os.html

[10] 日立ソフトウェアエンジニアリング株式会社, サン・マイクロシステムズ株式会社 “セキュアなインターネットサーバー構築に関する調査,”IPA/ISEC 第二回情報セキュリティ関連の調査・開発に関する公募.
URL=http://www.ipa.go.jp/security/fy14/contents/trusted-os/guide.html