安全なインターネットサーバ実現のための基盤ソフトウェア
― 不正アクセス被害の最少化を目指して ―

品川 高廣
東京農工大学工学部情報コミュニケーション工学科
shina @ cc.tuat.ac.jp

1　背景

近年のインターネットの普及に伴い，その基盤を担う存在であるインターネットサーバの役割が重要になってきている．インターネットサーバとは，WWW や電子メールなどインターネット上の様々なサービスを実際にユーザに提供しているコンピュータのことで，例えば Web サーバでは，ユーザが要求したページの URL を受け取って、指定されたページの内容を Web ブラウザに送り返すといった作業を行なっている．インターネットサーバは，ユーザがインターネットを利用する上で必要不可欠な存在であり、その信頼性・安全性を確保することは，一般ユーザが安心して利用できる IT 社会の実現にあたって極めて重要な要素である．

しかし現在のインターネットでは，インターネットサーバに対する不正アクセスが頻繁に発生しており，そのセキュリティ対策が重要な課題となっている．インターネットサーバが不正アクセスされると，例えばホームページが勝手に書き換えられたり，ユーザがサーバに送信した個人情報が漏洩したりするなど，様々な被害が発生する可能性がある．実際に，インターネットサーバに対する不正アクセスの危険性は IPA のセキュリティセンター [1] や CERT/CC [2] などにおいて頻繁に報告されており，深刻な問題となっている．このような不正アクセスが行なわれる根本的な原因はソフトウェアの欠陥であるが，残念ながらソフトウェアの欠陥を完全に無くすことは極めて難しい問題である．

2　目的

本プロジェクトの目的は，インターネットサーバに対する不正アクセスの被害を軽減するために，オペレーティングシステムによる強力な保護機構を開発することである．不正アクセスを完全に防ぐことは極めて難しい問題なので，本プロジェクトでは，仮に不正アクセスがおこなわれたとしても，その被害を最小限に抑えるための安全装置としての役割を果たす機構の開発を目標としている．本プロジェクトで開発した保護機構では，**サンドボックス**と呼ばれる手法を導入して，サーバソフトウェアを「箱」の中に閉じ込める．これにより，仮にサーバソフトウェアの欠陥を攻撃されて制御を乗っ取られたとしても，アクセス出来るのは箱の中の資源だけであり，箱の外に置かれた資源は確実に保護することができる．

このようなサンドボックス機構を通常のオペレーティングシステムで実現すると，オーバーヘッドが大きくなって性能が低下したり，様々なインターネットサーバに柔軟に対応できないなどの問題が生じる．本プロジェクトでは，我々が研究・開発してきたカーネルによる**細粒度保護ドメイン** [3] の技術を応用することにより，保護のオーバーヘッドを低く抑えつつ，柔軟な保護ができる機構を実現した．これにより，インターネットサーバを安全に実行できる環境を提供できるようになる．

3　開発の内容

本プロジェクトでの開発内容は，大きく分けると，(1) 細粒度保護ドメインの機構を備えたカーネルの開発，(2) 細粒度保護ドメインをベースにしたサンドボックス機構の開発，の 2 つに分けられる．本プロジェクトで開発した保護機構では，カーネルが提供する細粒度保護ドメインの機構を用いて，インターネットサーバが発行するシステムコールの内容を，サーバと同じプロセス内で動作する**参照モニタ**と呼ばれるプログラムで監視することによりサンドボックスを実現している．

3.1　細粒度保護ドメイン機構の開発

Linux 2.6.0 カーネルをベースとして，サンドボックスを実現する基盤となる細粒度保護ドメインをスクラッチから実装し直した．細粒度保護ドメインを実現する要素技術として実装した内容は，以下のとおりである．

プロセス内のメモリ保護機構　サーバプログラムを監視する参照モニタのプログラムをサーバプログラムから保護するために，プロセスの内部でのメモリ保護を提供する機構を実装した．実装には Pentium プロセッサに搭載されているセグメント機構を活用し，参照モニタとサーバプログラムを別々のセグメント内で動作させることでメモリ保護を実現している．また，参照モニタをユーザレベルで用意に実装できるようにするために，セグメントの構成を工夫してシステムコールの引数のチェックなどを効率よく行なえるようにしている．

高速な保護ドメイン切替機構　参照モニタによる監視のコストを抑えるために，インターネットサーバが動作する保護ドメインと参照モニタが動作する保護ドメインを高速に切り替える機構を実装した．具体的な保護ドメイン切り替えの実現方法としては，カーネル内でセグメントの定義を書き換えることで実装している．これにより，サーバプログラムの実行中は参照モニタに割り当てられたセグメントにアクセスできないようにして参照モニタを保護している．

保護ドメイン切り替えを高速に行なうために，切り替えを行なうコードは高度な最適化が施されている．例えば，従来のプロセス切り替えと比べてスケジューリングやページテーブルの切り替えなどの作業を一切行なわないため，プロセス切り替えの 10 分の 1 のコストで保護ドメイン切り替えを行なうことが出来る．切り替え作業を行なうコードは全てアセンブラで記述されており，余分なメモリアクセスの削減や L1 キャッシュの有効利用などの技術により，切り替えのオーバーヘッドを極限まで抑えている．さらに本プロジェクトでは，PentiumII 以降の CPU が備えている SYSENTER 命令を新たに採用し，カーネルモードへの切り替えコストを削減している．

システムコール横取り機構　参照モニタがサーバプログラムのアクセス内容をチェックできるようにするために，サーバプログラムが動作している保護ドメイン内で発行されたシステムコールを参照モニタでフック（横取り）するための機構を実装した．参照モニタでは、発行されたシステムコールの種類や引数などからシステムコールの発行を許可するかどうか決定する．この機構により，サーバプログラムが発行できるシステムコールを制限して，サンドボックスを実現することが出来る．

3.2 サンドボックス機構の開発

前節で説明した細粒度保護ドメインの機能を利用して，高速かつ柔軟なサンドボックスを実現するプログラムを実装した．実際には，サーバプログラムが発行するシステムコールを参照モニタでフックして，サンドボックスの外にあると判断した資源へのアクセスを禁止することにより，仮にサーバプログラムが乗っ取られたとしても，システムへの不正アクセスを防止することが出来る．

参照モニタは独立したモジュールとして ELF ファイルとして格納しておき，起動時に読み込んでサンドボックスとして組み込めるようにしている．読み込むファイルは起動時にオプションとして指定できるようになっており，ユーザが独自に実装した参照モニタを組み込むなど，必要に応じて柔軟に変更することが出来る．

サーバプログラムからは透過的にサンドボックスの構成処理（参照モニタの読み込みや細粒度保護ドメインの作成・割り当て等）を行なうために，サーバプログラムをプロセスのメモリ空間内に読み込む働きをする ELF ローダと呼ばれるプログラム (Linux では /etc/ld.so) を改変することでサンドボックスを実装した．これによって，サーバプログラムのバイナリを一切改変せずにサンドボックスを適用することが出来るようになった．

【起動例】

```
/lib/ld-sandbox.so -pm /lib/pm.so /usr/sbin/httpd
ld-sandbox.so: サンドボックスプログラム（プログラムローダを改変）
pm.so: 参照モニタプログラム
httpd: サーバプログラム (apache)
```

4 従来の技術との相違

従来から用いられている不正アクセスの防止手法としては，(1)WindowsUpdate 等のように脆弱性を修正するパッチを当てる方法，(2) パケットフィルタを用いてポート毎にパケット単位で不正アクセスを遮断する方法，(3) 侵入検知システムにより不正アクセスを事後に検出する方法，などがあげられる。本プロジェクトで採用したサンドボックスは，従来の手法と比べて以下のような利点がある．

未知の攻撃に対応できる: サーバソフトウェア全体を箱の中に閉じ込めるため，どのような手法でサーバソフトウェアを攻撃しても，箱の外の資源は保護できる．

運用中のサーバを保護できる: サーバプログラムが動作する環境を監視するため，サーバに送られるパケットの内容には依存せずに保護ができる．

攻撃による被害を未然に防止できる: 不正アクセスそのものを防止するため，被害を未然に防止することができる．

また、サンドボックスを実現する手法としては，研究レベルのものも含めると，(1) カーネルのみで実現する手法，(2) ユーザレベルだけで実現する手法，(3) 仮想マシンで実現する手法，などがある．これらの手法と比べると，本プロジェクトで開発したサンドボックスでは、以下のような利点がある．

柔軟性: 実際の保護ポリシーを決定する参照モニタをユーザレベルで実装できるため，サーバプログラムに特化した参照モニタを容易に作成できる．例えば，シンプルな実装で非常に強固な参照モニタを使用したり、非常に細かいアクセス制御が可能な高機能参照モニタを使用したりと、状況に応じて柔軟な保護を行うことができる。

互換性: 既存のサーバプログラムを改変する必要がないため、適用するサーバソフトウェアに制約がない．またソースコードが入手できない商用ソフトウェア等にも適用することができる．

性能: 保護によるオーバーヘッドが低いため，サーバの性能に対する影響を最小限に抑えられる．これにより，性能面が重要なサーバに対しても保護機構を適用しやすくなり，より多くのサーバの保護に利用することができる．

5 期待される効果

本プロジェクトの開発成果を利用することにより，安全なインターネットサーバを構築することが可能になる．直接的な効果としては，インターネットサーバを利用する組織において，不正アクセスによる被害の減少や管理コストの低下などが見込まれる．

また，本プロジェクトで開発した細粒度保護ドメインは極めて汎用的な機構であるため，単にインターネットサーバの保護にとどまらず，クライアント側の保護や新しいアプリケーションへの適用など，様々な応用が考えられる．また，利用目的としても，セキュリティ研究のための基盤から実際のシステムでの運用まで，様々な用途に用いることができる．従って，本プロジェクトの成果により，広い分野において全般的なセキュリティ向上に貢献することが期待される．

6 普及の見通し

開発成果は基本的にはフリーソフトとして Web で公開する予定であり，広範に使ってもらうことを目指している．公開のための作業は継続中であり，近日中に公開する予定である．既に何人かの方から使ってみたいという要望などをいただいており，ある程度の需要はあるものと見込まれる．

7 開発者名

品川 高廣 (東京農工大学工学部情報コミュニケーション工学科)
mailto:shina＠cs.tuat.ac.jp
http://www.sys.cs.tuat.ac.jp/~shina/mito/

参考文献

[1] IPA セキュリティセンター．http://www.ipa.go.jp/security/index.html

[2] CERT/CC. http://www.ipa.go.jp/security/index.html

[3] 品川高廣, 河野健二, 高橋雅彦, 益田隆司: 拡張コンポーネントのためのカーネルによる細粒度軽量保護ドメインの実現, 情報処理学会論文誌, Vol. 40, No. 6, pp. 2596–2606 (1999).