B-033

組込み OS 用暗号化ロードモジュール機能及びセキュアファイルサーバの試作

Prototyping of the Encrypted Load Module Capability and the Secure File Server for Embedded Operating Systems

望月 祐希† 野口 健一郎†
Yuuki Mochizuki Kenichiro Noguchi

1. はじめに

近年、組込み機器においても情報の漏洩や改竄等の対策が重要になっており物理的な脅威、ソフトウェアからの脅威に対してセキュリティを向上する必要がある．盗難等の物理的な脅威に対しては、補助記憶装置上のデータを全て暗号化することが望ましい．そこで、当研究室で開発中のセパレーションカーネルベースの安全な組込み OS に、カーネル及びパーティションプログラムの暗号化ロードモジュール機能と、パーティションプログラムが扱うデータを安全に管理するセキュアファイルサーバの試作を行った．

2. 背景

当研究室で開発中の OS は、高い信頼性と高いセキュリティの OS の実現を目指して、セパレーションカーネル方式を採用した[1]．セパレーションカーネルとは、複数の独立したパーティション空間とパーティション間の通信路を提供する OS 構成方式である．

しかし、ソフトウェアとしては高いセキュリティを持つが、物理的に対するセキュリティが未実装である．本研究ではこの点に注目し、デフォルトで補助記憶装置内のデータを暗号化するようにした．類似の機能を持つ OS として例えば、Chromium OS はデフォルトで補助記憶装置のデータを検証可能にし、ユーザのホームディレクトリを暗号化するようである[2]．

3. 暗号化ロードモジュール機能

3.1 概要

暗号化ロードモジュール機能は、カーネルロードモジュール、パーティションプログラムロードモジュール及びカーネルに必要なデータを暗号化したものを扱う．その概要を図 1 に示す．

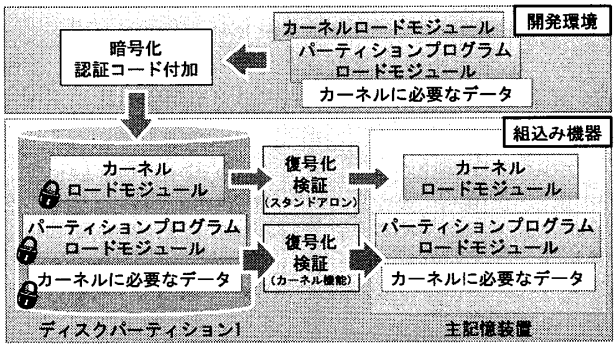

図 1 暗号化ロードモジュール機能の概要

† 神奈川大学大学院理学研究科情報科学専攻

開発環境で暗号化・認証コード付加されたカーネルロードモジュール、パーティションプログラムロードモジュール及びカーネルに必要なデータは、組込み機器で検証され主記憶装置上に格納する．

各々の検証でどれか一つでも失敗した場合、OS は起動せず停止する．

なお、試作システムで扱う全ての暗号方式は、Camellia 暗号を実装して使用した．Camellia[3]は、128bit のブロック暗号方式である．暗号利用モードは Counter with CBC-MAC (Cipher Block Chaining – Message Authentication Code)[4]を使用した．暗号化・復号化、検証・認証コード付加には、鍵と使い捨て乱数ノンスが必要となる．

3.2 カーネルロードモジュールの復号検証方式

カーネルロードモジュールは、試作したスタンドアロンの復号検証プログラムを用いて復号化・検証する．組込み機器の起動からカーネルが起動するまでの手順を図 2 に示す．

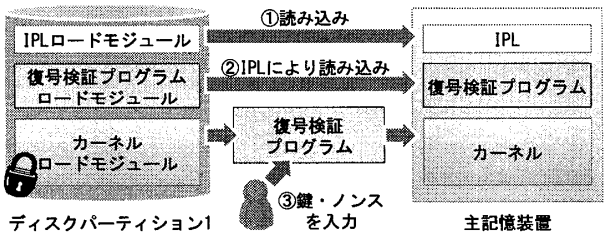

図 2 カーネルの復号検証手順

①起動すると IPL(Initial Program Loader)が主記憶装置に読み込まれる．②IPL によって、復号検証プログラムのロードモジュールが主記憶装置に読み込まれる．③復号検証プログラムは、システム管理者が入力する鍵・ノンスを用いて、カーネルロードモジュールを復号化・検証し、主記憶装置へ格納する．

使用した鍵・ノンスはそれ以降使用しないので主記憶装置上に残さないために、削除し、カーネルの起動を開始する．

3.3 パーティションプログラムロードモジュールの復号検証方式

パーティションプログラムロードモジュール及びカーネルに必要なデータであるパーティション間のメッセージ通信のポリシーファイル、鍵・ノンスデータは、カーネルの機能を用いて復号化検証し主記憶装置上に格納する．

（1）復号検証機能

復号検証機能は、パーティションプログラムロードモジュール、ポリシーファイル、鍵・ノンスデータを復号化・検証し、主記憶装置上へ格納する．その時に使用す

る、各々に対応する鍵・ノンスは、鍵・ノンス管理機能から受け取る．この関係を図 3 に示す．

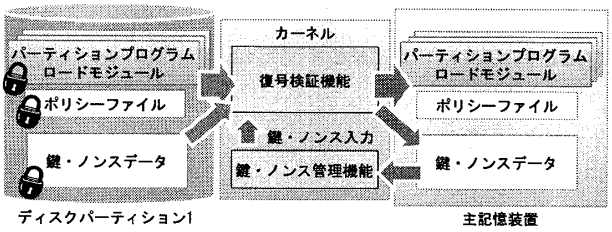

図 3　復号検証機能と鍵・ノンス管理機能の関係

（4）鍵・ノンス管理機能

鍵・ノンス管理機能は初めに、管理する鍵・ノンスデータを復号検証機能を使って取得する．鍵・ノンスデータ用の鍵・ノンスは、鍵・ノンス管理機能が既に持っておりこれを使用する．カーネルロードモジュールを復号化する鍵・ノンスは、鍵・ノンスデータに格納されるすべての鍵・ノンスと同等の価値を持つ．関係を図 4 に示す．

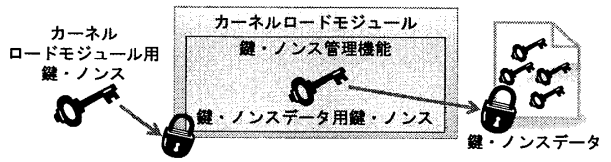

図 4　鍵・ノンスの関係

鍵・ノンス管理機能から一度取り出した鍵・ノンスは、それ以降必要ないので主記憶装置上に残さないために、削除する．

4．セキュアファイルサーバ

ファイルサーバは、ディスクサーバを通してディスクパーティション 2 を扱う．ファイルサーバは、暗号化認証コード付加機能と復号検証機能を持ち（カーネルの復号検証機能とは別のもの）、ファイルは 1 つ 1 つ、書き込み時は暗号化・認証コード付加を行い、読み込み時は復号化・検証を行い、ファイルの正当性を確認する．

また強制アクセス制御により、ユーザはファイルアクセスの設定を緩和することができないので、不正アクセスに対する安全性が高まる．

ファイルサーバは、ディスクパーティション 1 を扱う機能は持たない．また、カーネルもディスクパーティション 2 を扱う機能を持たない．これにより、パーティションプログラムによるカーネルの重要なファイルの改竄が困難になり、またカーネルの特権機能を減らすことができる．これを図 5 に示す．

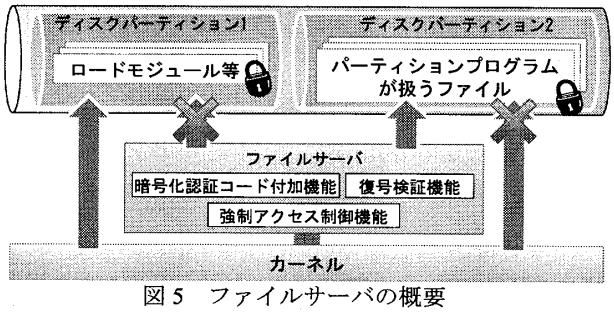

図 5　ファイルサーバの概要

ディスクパーティション 2 のフォーマットは、Ext2 を使用し、各々の inode にある拡張属性を利用して、その inode のファイルの鍵・ノンス、アクセス制御リスト、認証コードをファイルシステムが持つ鍵・ノンスによって暗号化・認証コード付加したものを拡張属性に格納した．

ファイル削除時は、ファイルの inode への参照を削除するという Ext2 の通常の方法に加えて、その鍵とノンスも削除する．これにより、データの復号が不可能になるので、実際のデータがまだディスク上に残り、それが漏洩の危険につながるという問題を解決することができる．

5．試作

本研究では以下の部分を試作した．

- ディスクパーティション 1 作成ツール
- ディスクパーティション 2 作成ツール
- Camellia プログラム
- 復号検証プログラム
- カーネルの復号検証機能、鍵管理機能
- ディスクサーバ
- ファイルサーバの一部の機能
- ディスプレイサーバ
- キーボードサーバ
- コンソール

ファイルサーバの暗号認証コード付加機能と復号検証機能、強制アクセス制御機能は試作中である．また、ファイルの操作 open、close、read を試作した．コンソールは、この三つの操作をテストするために試作した．

6．評価

本研究では、検証復号プログラムが正当であると仮定して、ディスクパーティション 1 で盗難等の漏洩が困難であること、改竄された OS が起動できない事を、試作したシステムを実機で動作させ、確認した．

カーネルロードモジュールの正当性を検証する為には、復号検証プログラム自信も改竄されていない正当性をもったものである必要がある．しかし、プログラムが実行を始めてから検証をしても無意味であり、自己検証ができない為、TPM (Trusted Platform Module)を利用する、書き換え不可能な補助記憶装置にプログラムを格納する等の対策が必要である．

7．結言

本研究では、組込み機器の補助記憶装置内のデータを暗号化する機能を、当研究室で開発中の組込み OS で実現するための試作を行った．今後の課題は次のものである．

（1）IPL・復号検証プログラムの正当性証明の解決

（2）ファイルサーバのファイル暗号化認証コード付加機能、復号検証機能、強制アクセス制御機能の追加

参考文献

[1]川守田 慶, 野口 健一郎: 形式手法を用いた組込み用 OS の試作 - B メソッドによる仕様検証実験 -, FIT2008

[2]Chromium OS, http://www.chromium.org/chromium-os

[3]RFC3713 — A Description of the Camellia Wncryption Algorithm, April 2004

[4]RFC3610 — Counter with CBC-MAC(CCM), September 2003