平成　２１年度　卒業論文概要			
所　　属	機械情報工学科	指導教員	光来　健一
学生番号	06237054	学生氏名	永田　卓也
論文題目	Cell/B.E.の SPE Isolation モードを用いた監視システム		

1. はじめに

近年、ネットワークを経由して個人の計算機が攻撃されるという事件が多発している。各ユーザーは AntiVirus などの監視ソフトウェアを利用してそれらの攻撃に備えるという対策をとるのが一般的である。しかし、監視ソフトウェアは OS の機能を利用して監視を行っており、攻撃によって OS が改竄されてしまった場合、監視ソフトウェアの実行結果を信用することはできなくなる。監視ソフトウェアの信頼性を向上させるためには OS が正しく動作していることを保証すべきであるが、OS はシステム全体を管理しているため、OS 自体が改竄されていないことを保証するのは難しい。

本研究では、Cell/B.E.の持つ SPE Isolation モードに着目し、SPE から OS カーネルを安全に監視するシステムを提案する。SPE Isolation モードを用いることにより、SPE 上で正しい監視プログラムが動作することを保証することができる。さらに、セキュリティプロキシから監視プログラムに定期的にハートビートを送ることで、監視プログラムが動作しているかどうかをチェックする。PS3 Linux 上に本システムを実装し、従来手法で監視プログラムを動かした場合と性能を比較した。

2. Cell/B.E.

Cell/B.E.は IBM、ソニー、東芝が共同開発したヘテロジニアス型マルチコアプロセッサであり、PlayStation 3 や CELL REGZA 等に使用されている。このプロセッサは制御系プロセッサコアである PPE と、演算系プロセッサコアである SPE によって構成され、それぞれのコアとハードウェアは EIB と呼ばれるバスで接続されている。図 1 に Cell/B.E.の物理構成を示す。

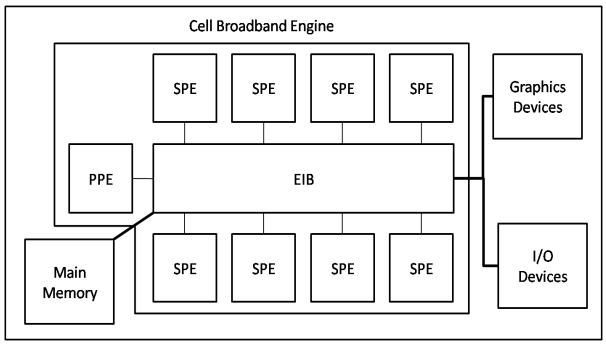

図 1．Cell/B.E.の物理構成

PPE は Cell/B.E.全体を制御することができるため、システム全体の管理を行う OS は PPE 上で動作する。OS カーネルのプログラムやデータはメインメモリ上に置かれる。一方、SPE には内部に Local Store（LS）と呼ばれるメモリ領域が存在し、DMA 転送を用いてメインメモリからプログラムをロードしたり、演算に必要なデータを取得したりする。

監視プログラムは OS が動作している PPE 上で動作させるのが一般的であるが、攻撃を受けて OS が改竄されてしまうと、監視プログラムも正常に動作しなくなる。監視プログラムを SPE 上で動作させれば、OS が動作している PPE からハードウェア的に分離されているため安全性は向上する。しかし、PPE は SPE を制御する機能を持つため、SPE で動いている監視プログラムを改竄したり、実行を停止させたりすることができてしまう。

3. カーネルメモリ監視システム

本研究では SPE Isolation モードを用いることで、監視プログラムを SPE 上で安全に動作させることができるシステムを提案する。さらに、Cell/B.E.搭載マシンの外部に置いたセキュリティプロキシから SPE 上の監視プログラムに定期的にハートビートを送ることによって、監視プログラムが動作していることをチェックする。監視プログラムの例として、メインメモリ上の OS カーネルの整合性をチェックするプログラムを対象とする。図 2 にシステム構成図を示す。

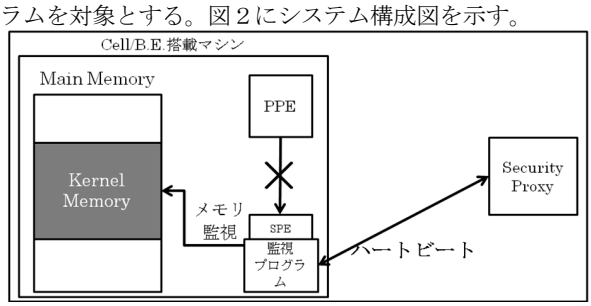

図 2．システム構成図

3.1 SPE Isolation モードによる安全な実行

SPE Isolation モードとは、SPE の持つ LS 領域に対し、PPE や他の SPE からのアクセスを禁止する機能である。LS 領域に対して外部からアクセス不可になるため、LS 内部にロードされたプログラムを実行中に改竄することはできない。その上、セキュリティプロキシとの間でハートビートを行うために LS 上に格納する必要がある暗号鍵を盗まれてしまうこともない。

また、SPE Isolation モードで動かすプログラムはコンパイル時にハードウェアが持つ鍵によって暗号化されており、LS にロードする際に Secure Loader によって復号化される。Secure Loader はプログラムの復号化と同時に整合性のチェックを行うため、攻撃者が改竄したプログラムは実行できない。

3.2 ハートビートによる実行のチェック

セキュリティプロキシは監視プログラムに定期的にハートビートを送り、監視プログラムが動作しているかどうかのチェックを行う。これは、SPE を Isolation モードで動作させたとしても PPE から SPE の実行を停止することはできてし

まうためである。ハートビートに対して監視プログラムから正しい返答がなかった場合には、監視プログラムが不正に停止させられたと判断し、セキュリティプロキシが Cell/B.E. 搭載マシンからのすべての通信を遮断する。

監視プログラムだけがハートビートに正しく返答できるようにするために、セキュリティプロキシは監視プログラムとの間の共通鍵を用いて暗号化したハートビートを送る。まず、セキュリティプロキシは乱数を生成し、それを共通鍵によって暗号化する。PPE 上の OS がネットワークからハートビートを受け取るとそのまま SPE 上の監視プログラムに渡される。監視プログラムはハートビートを復号化し、別の共通鍵で暗号化してセキュリティプロキシに送り返す。

3.3 カーネルメモリの監視

SPE 上の監視プログラムが DMA を用いてメインメモリにアクセスできるようにするために、SPE が持つ Segment Lookaside Buffer (SLB) にカーネルメモリのアドレスを登録する。SLB は仮想アドレスに実効アドレスを対応づけるためのテーブルである。さらに、アクセスするのに特権を必要とするカーネルのメモリ空間に SPE がアクセスできるようにするために、SPE の持つ Memory Flow Controller (MFC) の状態を変更する。

DMA 転送には一定の時間がかかるため、監視プログラムの高速化のために DMA 転送とメモリ内容のチェックを並列に行う。例えば、配列 A に対して DMA 転送を行うと、その DMA 転送が完了するまでは配列 A のデータを信頼することができない。そこで、本システムでは配列を 2 つ用意し、配列 A に対し DMA 転送を行っている間に、DMA 転送が完了している配列 B の内容をチェックする。

4. 実験

実験には PlayStation 3 の 80GB モデル、OS に Fedora 9 の Linux 2.6.27.25-78.2.56 を用いた。

4.1 監視プログラムの実行時間

OS カーネルが使用しているメモリ領域をメインメモリから SPE の LS に DMA 転送し、内容をチェックする監視プログラムの実行時間を測定した。図 3 に実験結果を示す。“SPE システム” が全体の実行時間であり、“SPE メモリアクセス” はその内の DMA 転送を行うのにかかった時間である。カーネルメモリ全体を監視するのにかかる時間は 8msec 程度であり、十分実用的である。

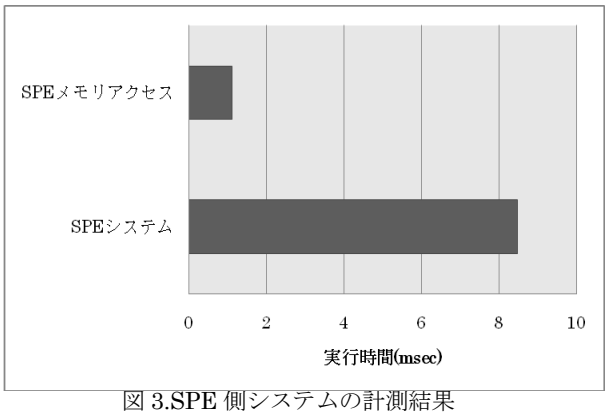

図 3.SPE 側システムの計測結果

比較のために、OS カーネルのサイズと同じ 12MB のメモリを確保して内容をチェックする PPE プログラムを作成し、実行時間を測定した。図 4 に実験結果を示す。“PPE システム” は全体の実行時間、“PPE メモリアクセス” はメモリのアクセスにかかった時間のみである。これら 2 つの実験の結果、SPE で監視プログラムを動作させるほうが、PPE で動作させるよりも高速であることが分かった。これは SPE による DMA 転送が PPE によるメモリアクセスよりも高速であるためである。

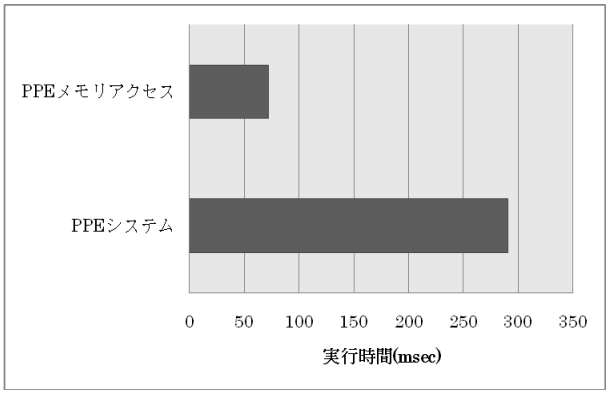

図 4.PPE 側システムの計測結果

4.2 カーネルメモリの整合性のチェック

OS カーネルのアドレス空間を 0x100000 毎に区切ってメモリの内容の和を求め、カーネルプログラムについてあらかじめ計算しておいた値と比較した。表 1 にアドレス毎の比較結果を示す。時間によって不変であった領域が○、そうでなかった領域が×である。

表 1. メモリ内部調査結果

先頭アドレス	中身が不変である
0x000000	○
0x100000	○
0x200000	○
0x300000	○
0x400000	×
0x500000	×
0x600000	○
0x700000	○
0x800000	×
0x900000	×
0xa00000	×
0xb00000	×

カーネル内部で不変であった領域はコード領域および読み込み専用領域であり、変化した領域はデータ領域であった。データ領域は刻一刻と変化するので、整合性をチェックする対象にする必要はない。

5. まとめ

本研究では Cell/B.E.の SPE Isolation モードを用いて OS カーネルを安全に監視することができるシステムを提案した。SPE Isolation モードによって正しい監視プログラムが実行されることを保証することができる。さらに、監視プログラムが動作しているかどうかチェックするためにセキュリティプロキシを用いた。今後の課題として、用いる暗号の強化、セキュリティプロキシでのチェックの強化や、ハートビートを SPE 単体で処理できるようにすることなどが挙げられる。