組込みシステム向け仮想化環境を用いた，汎用OSの信頼性向上のためのフレームワークの構築+SafeG-COMの拡張

三浦 功也[1,a)]　本田 晋也[1]　高田 広章[1,b)]

概要：本ポスターは，組込みシステム向け仮想環境である SafeG を用いた監視フレームワークと，SafeG が持つ通信機能である SafeG-COM の拡張についての発表を行う．具体的な内容としては，SafeG-COM 通信を多：多で行うための拡張方針や，監視フレームワークの全体像，また，フレームワークを構成する要素技術である，実行シーケンス監視機能などについて説明する．

キーワード：セキュリティ，フレームワーク，組込みシステム，リアルタイム OS

Proposal security flamework using Real-Time Embedded Virtualization and Extend SafeG-COM

Takuya Miura[1,a)]　Shinya Honda[1]　Hiroaki Takada[1,b)]

1. はじめに

組込みシステムに用いられるプロセッサの性能が向上するにつれ，組込みシステムの分野においても，従来のリアルタイム OS ではなく汎用 OS を使用したいという要求がある．しかし，組込みデバイスに汎用 OS を用いるためには，リアルタイム性と信頼性の面で不安が残るため，なかなか容易に適用できない現状がある．組込みシステム向け仮想化環境である SafeG を用いることで，リアルタイム性の問題を解決可能である．また，リアルタイム OS から汎用 OS の監視を実現することで，信頼性の向上も期待できる．しかし，これらの機能の実現が難しいという問題があり，監視システムを容易に実現するためのフレームワークの実現が求められている．

そのため本研究では，SafeG を用いたシステムにおいて，監視機能を容易に実現可能とすることを目的として，いくつかの監視機能を実装した．具体的には，監視機能実現のために必要な，両 OS で発生した割込み数の取得機能，汎用 OS のプロセスの実行シーケンスの取得機能を実現するためのプログラムの作成を行い，それらを容易に設定するためのフレームワークの提案を行った．

[1] 名古屋大学大学院 情報科学研究科
Furocho, Chikusa-ku, Nagoya-shi, Aichi, 464-0814, Japan
[a)] taku@ertl.jp
[b)] hiro@ertl.jp

2. 実行シーケンス取得機能

汎用 OS からリアルタイム OS にシーケンス番号を送信し，リアルタイム OS からその番号を確認することで，正しく処理が流れているかを確認することができる．この機能を用いて，汎用 OS 既存の監視システムの動作を確認することで，そのシステムが監視している動作は正しく動作していることを保証できる．本機能では，シーケンス番号送信時の振る舞いとして，通知方式と非通知方式の二つを用意した．通知方式では，汎用 OS からシーケンス番号を送信する度に，リアルタイム OS に切り替える．非通知方式では，汎用 OS からシーケンス番号を送信しても，用意されたバッファがいっぱいになるまではリアルタイム OS に切り替えることなく，汎用 OS の処理を継続する．シーケンス番号を保存の際は，同時にシーケンス番号の送信時間や，送信元のアドレスを保存しておき，リアルタイム OS からはその値を基にして監視機能を実装する．

3. 割込み数取得機能

近年の汎用 OS のセキュリティソフトの中には，システムの動作の変化を確認することにより，攻撃の検知を行うものがある．SafeG を用いたシステムでは，発生した割込みは全て SafeG を経由するため，単位時間あたりの割り込み要因ごとの割り込み回数を確認することができる．これを確認することにより，設計したとおりに割込みの回数

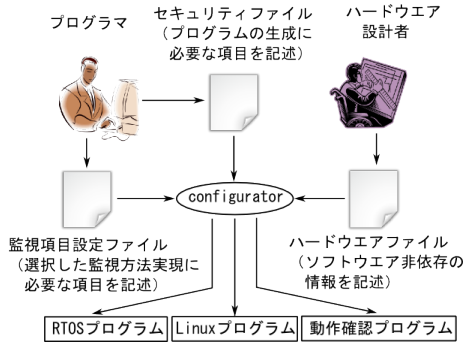

図 1 フレームワークの全体像

が入っているか，また，意図した割込みのみが入っているかということを確認することができ，振る舞い検知型のセキュリティソフトと同様の監視を安全に行うことができる．本機能では，割込み発生時に，割込みが発生したコアの，割り込み要因に対応したカウンタを上げる．割込み発生回数の記憶に関して，特定の割込み番号のカウントを個別に停止する機能は設けない．理由としては，全ての割込み番号を保存しても，メモリ使用量がそれほど増加しない点，カウントがオーバーフローしても 0 に戻るため，監視機能の動作に影響を与えない点，条件判断によるオーバーヘッドの増加を防ぐ点が挙げられる．

4. フレームワークの概要

本研究で提案するフレームワークの概要を，**図 1** に示す．ハードウエア設計者がハードウエアファイルを作成する．このファイルには，RAM の大きさなどのソフトウエア非依存の情報を記述する．アプリケーションプログラマは，使用する監視機能をセキュリティファイルに記述し，同時にシーケンス番号の最大値などの，監視機能実現に必要な項目について別途設定ファイルに記述する．これらの情報を基にコンフィギュレータを動作させることで，監視機能を持ったリアルタイム OS のアプリケーションと，汎用 OS のライブラリ機能，動作確認を行うためのプログラムを自動的に生成する．

5. 評価

シーケンス監視機能と割込み数取得機能に必要な実行時間の測定を行った．評価には，Cortex-A9 MPCore を搭載した FPGA ボードで行った．プロセッサの周波数は 800MH z であり，1Gbyte のメモリを搭載している．同時に動作させる OS は，安全系の OS，非安全系の OS ともに TOPPERS FMP カーネルを使用した．

5.1 割込み番号のカウンタを上げるまでの時間

割込みが発生した時に，該当する割込みの発生回数のカウントアップを行ために必要な時間を 10 万回測定し，その結果を 0.1us 単位で正規化した．その結果，最頻値は 0.1us〜0.2us であり，その割合は 99.9%であった．また，最悪実行時間は 0.4us〜0.5us であった．

5.2 シーケンス番号の送信を通知しない場合の実行時間

シーケンス番号の通知を行わない場合に，番号送信後に非安全系の OS に戻ってくるまでの時間を 1000 回測定し，0.5us 単位で正規化した．その結果，最頻値は 1.5us〜2.0us の間であり，その割合は 96.4%であった．また，最悪実行時間は 2.0us〜2.5us の間であった．

5.3 シーケンス番号の送信を通知する場合の実行時間

シーケンス番号の通知を行う場合に，番号送信後に非安全系の OS に戻ってくるまでの時間を 1000 回測定し，0.5us 単位で正規化した．その結果，最頻値は 1.5us〜2.0us の間であり，その割合は 81.9%であった．また，最悪実行時間は 2.5us〜3.0us の間であった．

5.4 考察

SafeG が実行する OS を切り替えるために必要な実行時間は 1.3us〜1.7us である場合がほとんどであった．[*1]．そのため，5.2 節と 5.3 節の結果のうち，今回の機能追加におけるオーバーヘッドは，それぞれ 0.3us，0.5us 程度だと考えられる．いずれの場合も，実行時間の増加は抑えられていると言える．また，割込み番号のカウントアップに関しても，シーケンス番号送信のオーバーヘッドよりも十分小さく，十分抑えられていると言える．

6. まとめ

本研究では，組込みシステム向け仮想化環境である SafeG を用いて監視機能を実現するために必要なフレームワークの提案と，それに必要な機能である割込み数取得機能と，実行シーケンス取得機能を作成した．しかし，現状では最低限の機能しか実現できていない．また，現在は非信頼系の OS として汎用 OS を適用していないため，実行シーケンスの監視におけるタイムウインドウ制御の機能や，ユーザモードからの呼び出し，実際の汎用 OS アプリケーションの監視，汎用 OS を利用した運用などが今後の課題である．

7. 参考文献

参考文献

[1] 太田貴也, Daniel Sangorrin, 一場利幸, 本田晋也, 高田広章“組込み向け高信頼デュアル OS モニタのマルチコアアーキテクチャへの適用”, 情報処理学会第 117 回 OS 研究会,(2011-4)
[2] 中島健一郎, 本田晋也, 手嶋茂晴, 高田広章“セキュリティ支援ハードウエアによるハイブリッド OS システムの高信頼化”, 情報処理学会研究報告 EMB, 組込みシステム, pp.1-7 (2008-11)

[*1] 同環境で事前に計測し取得した値