「仮想マシンモニタを用いた割り込み処理のデバッグ手法」の要約

2014/6/6

藤田将輝

1　はじめに

仮想マシンを用いた割り込み処理のデバッグ支援環境の構成を理解するため，情報処理学会研究報告である「仮想マシンモニタを用いた割り込み処理のデバッグ手法」[1] を読解した．本資料ではこの論文の要約を示す．

2　目的

OS の複雑化，多機能化に伴い，OS 内部に存在するバグが増えている．このため，OS のデバッグが重要になっている．OS のデバッグは困難であり，その理由の一つに非同期処理である，割り込みの再現がある．非同期処理とはデータを転送する際に，送信側と受信側のタイミングの一致を気にせずにデータをやり取りする処理である．割り込み処理は非同期的に発生し，常に同じタイミングで発生するとは限らないため，再現が難しい．本研究では割り込みのタイミングを制御可能にすることで OS のデバッグを支援する手法を提案する．提案手法では仮想マシンモニタを利用し，割り込みのタイミングを制御する．

3　割り込み

割り込みは，どのタイミングでどの割り込みが発生しても正しく処理しなければならない．まず，割り込みは適切に禁止/許可されなければならない．割り込みが発生することにより，処理結果に影響を受ける関数や，プログラムがあるためである．また，どのようなデータを受け取っても適切に処理しなければならない．受け取ったデータがどのようなものでもデータ構造を壊したり，NULL ポインタを参照したりしてはならないためである．

4　提案手法

4.1　概要

提案手法では使用者が割り込みを発生させるコード位置と割り込みの種類を指定する．指定したコード位置で任意の割り込みを発生させる．この手法を用いることで，任意のタイミングで任意の種類の割り込みを意図的に発生させることができる．

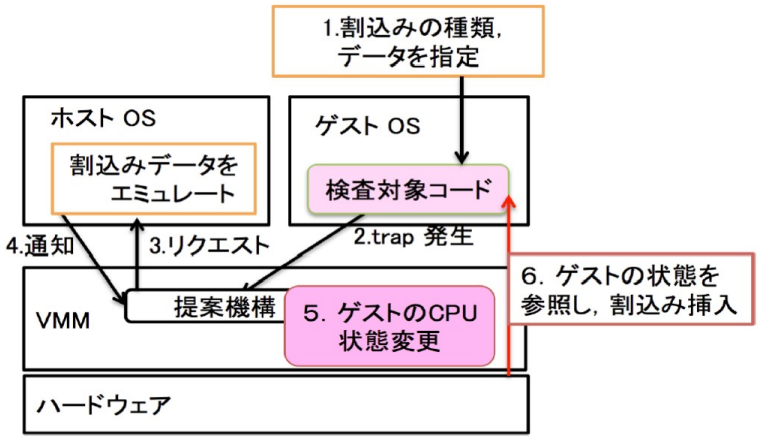

図 1　提案手法における割り込み処理の流れ

4.2　割り込み挿入の方法

提案手法を実現するために，仮想マシンモニタ (VMM) による仮想環境を用いる．仮想環境では仮想マシンのイベントを VMM がエミュレートする．エミュレートとは仮想マシンの動きを再現することである．提案手法ではデバッグ対象 OS をゲスト OS として動作させる．VMM を用いることで意図的にゲスト OS に割り込みを発生させることができる．提案手法における割り込み処理の流れを図 1 に示し，以下で説明する．

(1) デバッグ対象 OS の割り込みを挿入したいコード位置に trap を発生させるコードを挿入し，ゲスト OS として動作させる．なお，trap には割り込みの種類とデータを指定しておく．
(2) (1) で指定したコード位置でゲスト OS が trap を発生する．
(3) VMM がホスト OS へ割り込みデータを準備するリクエストを渡す．
(4) ホスト OS が割り込みデータを作成し，VMM にデータ作成完了を通知する．
(5) VMM がゲストの CPU の状態を書き換える．なお，ゲスト OS の CPU の状態は VMCS というデータ領域で管理されており，実際にはこれを書き換える．
(6) 処理がゲスト OS に移り，CPU がデータ領域を参照し，割り込みが挿入される．

これにより，指定したコード位置に割り込みを挿入させることができる．

5　実装

5.1　概要

提案手法を仮想マシンモニタ Xen4.1.0 に実装した．Xen ではゲスト OS を domU，ホスト OS を dom0 と呼ぶ．提案手法の実装イメージを図 2 に示し，以下で説明する．

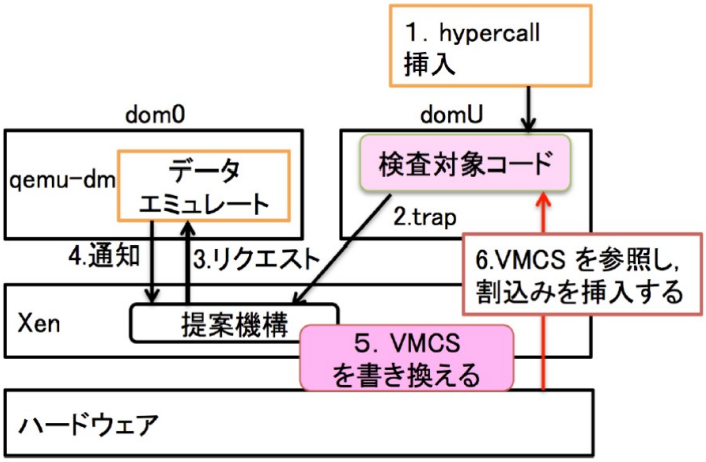

図 2　実装図

(1) domU の割り込みを発生させるコード位置に hypercall を挿入する.
(2) 挿入した hypercall が trap を生成する.
(3) Xen が dom0 に割り込みデータ作成のリクエストを渡し，dom0 がデータを作成する.
(4) dom0 は Xen に割り込みデータ作成完了を通知する.
(5) Xen が VMCS を書き換える.
(6) CPU が VMCS を参照し，割り込みを挿入することで，domU で仮想的に割り込みが発生する.

Xen では，割り込みデータを qemu-dm で処理している．そこで，qemu-dm に変更を加え，データを準備する機構を作成する.

5.2　割り込み挿入位置の指定

ゲスト OS の割り込みを挿入させたいコード位置にハイパーコールである VMCALL を挿入する．VMCALL はレジスタの中身を引数として渡すことができ，割り込みの種類を指定する．また，送信する割り込みデータもここで指定する.

5.3　割り込みの挿入

割り込みの挿入は Xen が VMCS を書き換えることで行う．具体的には VMCS の VM-entry control field という領域における，VMEntry Controls for Event Injection という領域を書き換える．これにより,VMM から domU に処理が移る際に割込みを仮想的に挿入できる．dom0 がリクエストを受けてから割り込み挿入までの流れを以下に示す.

(1) dom0 がデータを準備するリクエストを受け取り，割り込みデータをエミュレートする関数を呼び出す.
(2) 割り込みデータをエミュレートする関数により，割り込みデータを domU のドライバ用のデータ

に変更する．

(3) 関数の処理が完了すると，Xen に割り込みデータの作成完了を通知する．

(4) Xen が，通知を受け取り，VMCS を書き換え，操作を domU に移す．

(5) 操作が移る際，CPU が VMCS を参照し，仮想的に割り込みが発生する．

6 まとめ

本研究では割り込みのタイミングを意図的に制御可能にすることによりデバッグを支援する手法を提案した．また，Xen4.1.0 上に割り込みを意図的に挿入する機構を作成した．domU の検査対象コードから呼び出されるハイパーコールをうけ，Xen が VMCS を書き換える仕組みを実装した．また，qemu-dm 上に，割込みのデータを準備する機構を作成した．この機構を利用してタイマ割り込み，キーボード割り込み，ネットワーク割り込みの挿入を行えるようにした．現在挿入できている割り込みは，タイマ割り込み，キーボード割り込み，ネットワーク割り込みのみである．これ以外の割り込みについては対応できていない．また，現在の実装ではハイパーコールをコードに直接埋め込んでいるため，利用者が入れるべき割り込み場所，種類を予想して利用しなければならない．

7 おわりに

本資料では「仮想マシンモニタを用いた割り込み処理のデバッグ手法」[1] を要約した．仮想マシンを用いたデバッグ支援手法の目的と動きを理解した．また，割り込みにおけるバグの例を知ることができた．仮想化を用いた割り込みのデバッグ環境と Mint を用いたデバッグ支援機構と照らし合わせ，デバッグ支援機構の理解を深める．

参考文献

[1] 宮原俊介，吉村剛，山田浩史，河野健二:仮想マシンモニタを用いた割り込み処理のデバッグ手法，情報処理学会研究報告，Vol.2013-OS-124，No.6，pp.1-8(2013)