ハイパーバイザBitVisorの保護ドメインを用いた高セキュアなコンテナの研究

肥沼 健[1]　並木 美太郎[1]

1. はじめに

コンテナ技術による仮想化を利用したアプリケーションの開発や運用が一般的となってきている．コンテナはホスト OS のカーネルを共有するためホストの権限が不正に獲得され他のコンテナへ攻撃できる可能性があるためセキュリティ対策としてクラウド上ではテナントごとに仮想マシン（VM）を起動させその上でコンテナを構築している．セキュリティを確保するために，コンテナを構築するための余裕を持ったリソースで VM を起動させることが必要であり，さらにはその VM の管理をしなくてはならない問題が発生する．

本研究ではこうしたセキュリティに起因するコンテナの利用時の非効率性を解消するために，ハイパーバイザである BitVisor の保護ドメインを利用したコンテナ構築の手法を提案する．

2. 研究の課題

コンテナのセキュリティを確保するため，従来の仮想化基盤の拡張を行うことで VM の分離特性を備えたコンテナ構築手法が提案されている [1]．また従来の仮想化基盤は軽量な VM を高速に起動させることは不得意であるとして，Linux や BSD を仮想化のバックエンドとし，システムコールの発行制限をしながら OS 上のプロセスとして起動させる手法も提案されている [2]．いずれも従来の大規模な仮想化基盤としてのソフトウェアあるいは仮想化を目的とすることに対しては多くの機能を備えた巨大な汎用 OS を利用し，これらは脆弱性も毎年報告されている．

コンテナに特化したものであり，仮想化を行う基盤自体のセキュリティに信頼をおける高セキュアな仮想化の実行基盤が必要である．

[1] 東京農工大学

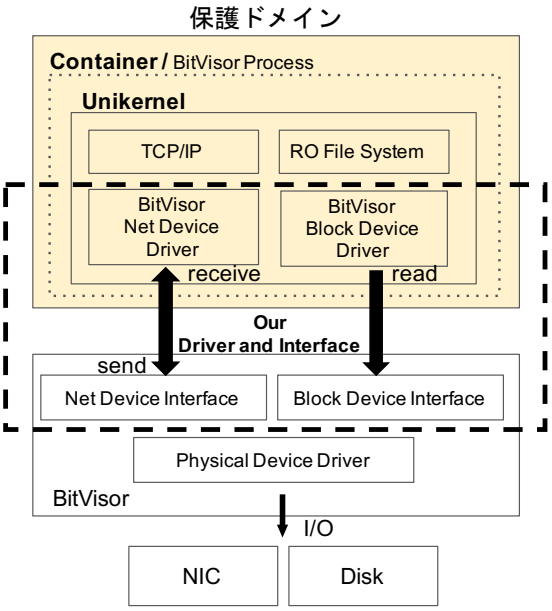

図 1　全体構成図

3. 提案

本研究では，BitVisor の Root Mode の Ring 3 の実行である保護ドメインを利用して Unikernel を動作させることでコンテナを構築する手法を提案する．BitVisor が専用のインターフェースを提供し，Unikernel がこのインターフェースを利用して保護ドメインで動作する．本研究ではこれをコンテナとする．

Root Mode の実行であるため仮想化支援機構による CPU のモード遷移をすることなく高速化が期待できる．しかし短所として I/O 命令を仮想化機構によって容易にトラップできなくなるため，セキュリティを確保することが難しい．そこで I/O の監視と仲介，メッセージ I/F によるハイパーバイザのコア機能から分離された実行空間である保護ドメインを備える BitVisor を利用することで Unikernel を

表 1　BitVisor ハイパーバイザコール

ハイパーバイザコール	機能
bv_net_send	BitVisor へパケットを送る
bv_net_receive	BitVisor からパケットを受け取る
bv_block_write	BitVisor へブロックデータを書き込む
bv_block_read	BitVisor からブロックデータを読み込む
bv_get_time	BitVisor から時刻を取得する

セキュアに動作させることができる．

全体の構成を図 **1** に示す．コンテナではネットワークプロトコルスタック，読み込み専用のファイルシステム，BitVisor のインターフェースを満たすドライバを備えた Unikernel を利用する．BitVisor ではネットデバイス，ブロックデバイスとしてのインタフェースを定義して，仮想デバイスとする．

3.1　BitVisor

Unikernel に対してブロックデバイスとネットワークデバイスのインターフェースを提供する．これは BitVisor ハイパーバイザコールとして多機能であることよりも最小な構成で定義をして（表 **1**）かつ BitVisor のコアの部分を大幅に変更することなく最低限の追加である．これらのハイパーバイザコールは BitVisor が I/O の監視と制御のために備えているドライバを利用して最終的に物理デバイスへの I/O となる．

3.2　Unikernel

Unikernel はライブラリ OS とアプリケーションが一体となり単一のアドレス空間で動作する特徴があり，起動時間も短く，保護ドメインで動作させるコンテナとの適性が高い．Unikernel のアプリケーションはカーネルと同一の 1 プロセスである．1 つの Unikernel は保護ドメインでは 1 プロセスでの動作となり，保護ドメイン上では複数の動作となる（図 **2**）．つまり保護ドメインで複数のコンテナが動作する．Unikernel は CPU の仮想化支援機構を利用した仮想化基盤上の動作を前提として実装され，各プラットフォームに合わせて最低限のドライバを備えている．本研究では BitVisor をプラットフォームとする軽量なドライバを追加する．Unikernel に追加するドライバは，表 **2** に示す最低限の機能を実装する．

コンテナとしての Unikernel の動作はライフサイクルが短く，VM と異なり起動後に構成を整えることはせず，決まった定義及び初期値によって起動する．このことはファイルへの書き込みができることの重要性が低いことを意味し，重要性の低いファイルへの書き込みは利用しない．

4.　おわりに

我々は現在，BitVisor のハイパーバイザコールの動作を確認できる段階まで実装して Unikenel のドライバを追加

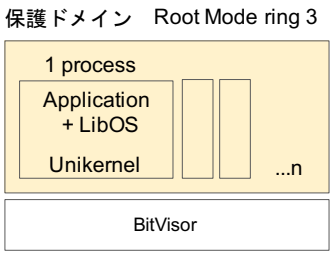

図 **2**　コンテナの複数動作

表 2　BitVisor 向け Unikernel デバイスドライバの機能

名称	デバイス	機能
create_packet	ネットワーク	パケットを生成する
transmit	〃	パケットを送る
receive_packet	〃	パケットを受け取る
block_size	ブロック	ブロックの大きさを返す
write	〃	ブロックデータを書き込む
read	〃	ブロックデータを読み込む

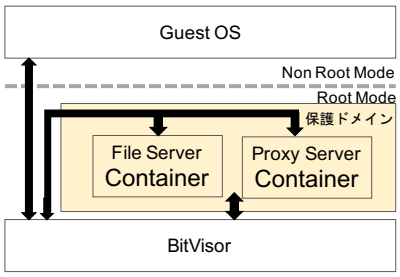

図 **3**　ゲスト OS とコンテナの連携

している．今後は実装を完了させ評価をする予定である．

また今後の展望として，保護ドメインでコンテナを構築して動作させ評価を行うだけにとどまらず，Non Root Mode での実行で BitVisor 上で動作するゲスト OS とコンテナが連携する図 **3** の機構も検討している．この機構の実現により BitVisor で動作するコンテナがゲスト OS よりも特権レベルが高いという優位性を持ちながら連携をすることで，例えばゲスト OS と通信するファイルサーバやプロキシサーバなどをセキュアにかつコンテナの単位であたかも機能を追加するような形で提供することができると考えている．

参考文献

[1] Shen, Z., Sun, Z., Sela, G.-E., Bagdasaryan, E., Delimitrou, C., Van Renesse, R. and Weatherspoon, H.: X-Containers: Breaking Down Barriers to Improve Performance and Isolation of Cloud-Native Containers, *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '19, p. 121–135 (2019).

[2] Williams, D., Koller, R., Lucina, M. and Prakash, N.: Unikernels as Processes, *Proceedings of the ACM Symposium on Cloud Computing*, SoCC '18, p. 199–211 (2018).