# 言語処理系を組み込んだハイパーバイザによるネットワーク監視・制御のためのSDN基盤に関する研究

尾内 智哉[1,a)]　並木 美太郎[1,b)]

## 1. はじめに

SDN をネットワークセキュリティの強化に応用する方法が提案されている．しかし, コントロールプレーンの SDN コントローラはデータプレーンを集中制御するため, 単一障害点になる．そこで SDN コントローラへの DoS 攻撃の緩和や, Northbound API へのアクセスを制限する行う方法が提案されている．しかし, これらの対策は SDN コントローラよりも高い特権レベルで動作する OS やハイパーバイザが危殆化すると無効化される．OS は新しい機能の追加やハードウェアの対応のためにコードは肥大化し続けており攻撃層は大きく, 信頼できない OS もある．また, OS のバグが発見されてから数週間から数ヶ月経たないと, バグの本当のセキュリティへの影響が発見されないことがよくあり, パッチが作成され, 適用されるまで時間がかかることがある [1]．ハイパーバイザはデバイスのエミュレーションに関するバグが多く報告されており, アプリケーションの実行に必要のない処理に起因するバグも発生する [2]．

## 2. 目標

本研究では仮想化基盤として小さく必要最低限の機能を備えたハイパーバイザの保護ドメインでゲスト OS に依存せずにセキュアにネットワーク監視・制御を行う SDN 基盤を提案する．保護ドメインを使用することでハイパーバイザのコアと分離された低く特権レベルでセキュアにネットワーク監視・制御を行う．ネットワーク監視・制御は, スクリプトで記述したルールによってパケットフィルタリングをすることでネットワークを監視し, SDN によりプログラマブルにネットワークを制御することでネットワークセキュリティを強化する．

[1] 東京農工大学
Tokyo University of Agriculture and Technology
[a)] s219821x@st.go.tuat.ac.jp
[b)] namiki@cc.tuat.ac.jp

## 3. 設計

本研究で提案するシステムの構成を次の図 1 に示す．

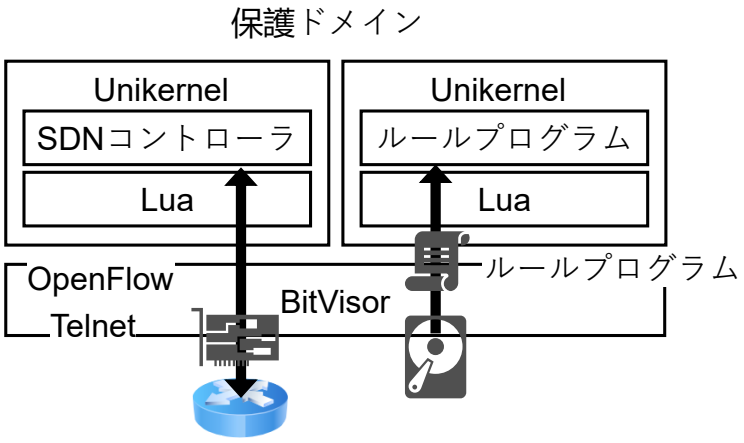


**図 1**　提案システムの構成

軽量なハイパーバイザの BitVisor の保護ドメインにスクリプト言語処理系を組み込み, スクリプトの高い表現力を活かしてスクリプトで記述したルールプログラムによって SDN によるネットワークの制御とゲスト OS のネットワークのフィルタリングによるネットワークの監視を行いクライアント PC とサーバを保護する．小さく必要最低限の機能を備えたセキュアなハイパーバイザを使用することでハイパーバイザへの攻撃を軽減し, OS に依存せずに信頼できない OS がある場合でも安全にプログラムを実行することができる．

### 3.1 保護ドメイン上で動作する SDN コントローラ

肥沼らが提案した, ハイパーバイザによる高セキュアなコンテナの実行基盤上で SDN コントローラを実行する．コンテナとして Unikernel の IncludeOS を使う．この Unikernel にスクリプト言語処理系の Lua を移植して OpenFlow プロトコルを扱う SDN コントローラが動作する SDN 基盤を

構築する.

SDN コントローラとルールプログラムは別々の保護ドメインで実行し保護ドメインが持つメッセージ I/F を通してルールプログラムで分析するパケットや分析結果の送信を行う. SDN コントローラとルールプログラムを実行する保護ドメインではメッセージキューを持ち, ノンブロッキングな通信を行う. これにより SDN コントローラはルールプログラムによってブロックされることがなくネットワークを制御できる.

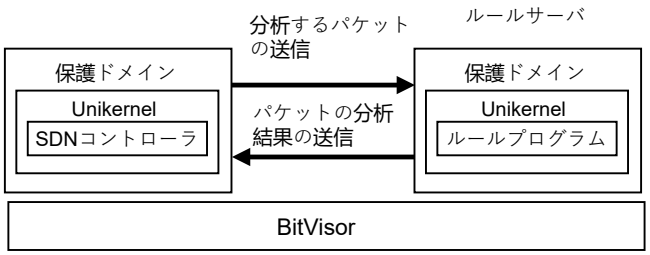


**図 2** 保護ドメインで動作する SDN コントローラとルールサーバ

SDN コントローラは Openflow を扱えないルータとも連携し, SDN でネットワークを制御する情報をもとにルータにもネットワーク制御のためのコマンドを送信して適用する.

### 3.2 ルールプログラムによるパケットの分析

ルールプログラムでは次の表 1 に示す.

**表 1** ルールプログラムで使用する Unikernel 用の Lua API

| API | 機能 |
|---|---|
| send_msg(desc, data, size) | desc で指定したディスクリプタを通してデータを送信する |
| dequeue_packet() | SDN コントローラから送信されてキュ-に格納されたパケットを取得する |
| send_result(ipaddr, port) | ルールプログラムによる分析結果を SDN コントローラへ送信する |
| dequeue_rule_result() | ルールプログラムから送信されてキュ-に格納された分析結果を取得する |

ルールプログラムの例として HTTP の通信を分析し, web アプリケーションの脆弱性の OS コマンドインジェクションを検査する例を次に示す. このルールでは正規表現で与えたパターンが HTTP の通信に含まれるかパターンマッチングによって検査する. OS コマンドインジェクションを検出したら指定した IP アドレスとポート番号を持つホストからの通信を遮断するという結果を SDN コントローラに送信する.

**プログラム 1** Lua で記述した HTTP の通信を分析するルール

```
1  while true do
2    eth_data = dequeue_packet()
3
4    if http_data(eth_data) then
5      request, header, body =
6        parse_http_request(eth_data)
7      -- OS コマンドインジェクションを検出するパターン
8      pattern = ".*;%scat%s/etc/passwd.*"
9      ret = string.find(request .. body, pattern)
10
11     if ret ~= nil then
12       send_result(
13         get_ip_src(eth_data),
14         get_dst_port(eth_data))
15     end
16   end
17 end
```

## 4. 評価

SDN コントローラとルールプログラムを連携させて実行することによる SDN コントローラのオーバーヘッドを計測した結果は次の表 2 の通りである. ルールプログラムの実行によるオーバーヘッドは (0.655-0.265) / 0.265 = 1.472 で 147%であり, SDN コントローラは分析のために 1 パケットずつルールプログラムへパケットを送信するが低いオーバーヘッドでパケットを分析することができた.

**表 2** ルールプログラムの実行によるオーバーヘッド

| 実行したプログラム | 実行時間 (ms) |
|---|---|
| SDN コントローラとルールプログラム | 0.655 |
| SDN コントローラ | 0.265 |

## 5. おわりに

本論文では BitVisor の保護ドメインを使って, SDN でネットワークセキュリティを強化し, クライアント PC とサーバを保護するための高セキュアな SDN 基盤を実現した. 今後は Unikernel のマルチスレッド対応により, 様々なアプリケーションをで実行できるようにしてネットワーク監視・制御に活用したい.

### 参考文献

[1] Arnold, Jeff, et al. ”Security impact ratings considered harmful.” arXiv preprint arXiv:0904.4058 (2009).
[2] Perez-Botero, Diego, Jakub Szefer, and Ruby B. Lee. ”Characterizing hypervisor vulnerabilities in cloud computing servers.” Proceedings of the 2013 international workshop on Security in cloud computing. 2013.