OpenFlowを用いた攻撃者遮断システムの提案と評価

下川 大貴[1]　小刀祢 知哉[1]　池部 実[2]　吉田 和幸[3]

概要：我々は，インターネットから学内ネットワークへ送信されるパケットを監視することで攻撃者を検知し，通信を遮断する「不正通信検知システム」を開発・運用してきた．不正通信検知システムで検知した攻撃者は，アクセスコントロールリスト (ACL) や，経路制御を用いて遮断していた．しかし，これらの方法にはいくつかの問題点がある．そこで我々は従来システムにおける，攻撃者の遮断手法の問題点を改善することを目的に，OpenFlow を用いた攻撃者遮断手法を提案・開発してきた．本論文では，Web サーバに対する DoS 攻撃を想定し，実験環境を構築した．また，実験環境上で OpenFlow スイッチを用いて攻撃者を遮断し，性能評価をした．OpenFlow を用いた攻撃者遮断手法の遮断効果，遮断時の他の通信への影響について考察し，OpenFlow を用いた攻撃者遮断手法の有効性を示す．

A proposal for the attacker blocking system using the OpenFlow and its evaluation

Shimokawa Daiki[1]　Kotone Tomoya[1]　Ikebe Minoru[2]　Yoshida Kazuyuki[3]

1. はじめに

インターネットの発展に伴い，ネットワークを通して様々な情報がやり取りされている．Web ページの閲覧や電子メールなどのコミュニケーション手段に留まらず，インターネット上での行政手続やクレジットカード番号を利用した電子決済など公共性の高いサービスも提供されている．そのため現在では，ネットワークは社会的基盤の一つとして生活に不可欠な存在になっている．インターネットを通じて，多様なサービスが利用されている一方で，OS やプログラムの脆弱性を利用し不正なプログラムを実行し，サーバへ不正にアクセスするなど，さまざまな脅威が存在する．警察庁の「インターネット観測結果」[1] によると，シグネチャを用いて検知した不正侵入等の行為の件数は、1日・1IP アドレス当たり 21.4 件で、前期と比較して 3.0 件（16.2 %）増加している．我々は，インターネットから学内ネットワークに対する不正通信を検知するため，送信されるパケットを監視し，TCP スリーウェイハンドシェイクの手順と異なるパケットを送信した攻撃者を検知し，遮断する「不正通信検知システム」[2][3] を開発・運用してきた．従来，不正通信検知システムにて検知した攻撃者の遮断には，アクセスコントロールリスト (ACL) や，経路制御を用いてきた．しかし，両者にはそれぞれ問題点がある．ACL を用いた手法では，攻撃者登録件数の上限や LAN スイッチの設置場所，経路制御を用いた手法には，攻撃者宛の応答パケットを遮断するため，攻撃者のパケットが学内を通過するという問題点がある．

そこで我々は，従来システムにおける攻撃者の遮断手法の問題点を改善することを目的に，OpenFlow を用いた攻撃者遮断手法 [4] を提案・開発してきた．我々は，これまでに OpenFlow を用いた攻撃者遮断手法の動作検証をするために，仮想環境で実験してきた．仮想環境での実験において，不正通信検知システムからメッセージ (攻撃者 IP アドレス) を受信した場合，OpenFlow コントローラから OpenFlow スイッチに対して，送信元 IP アドレスを識別して破棄するフローエントリを FlowMod で書き込み、攻撃者を遮断できることを確認した．

1 大分大学大学院工学研究科知能情報システム工学専攻
Course of Computer Science and Intelligent Systems, Graduate School of Engineering, Oita University
2 大分大学工学部知能情報システム工学科
Department of Computer Science and Intelligent Systems, Faculty of Engineering, Oita University
3 大分大学学術情報拠点情報基盤センター
Center for Academic Information and Library Services, Oita University

そこで本論文では，Web サーバに対する DoS 攻撃を想定し，環境を構築した．そして，OpenFlow スイッチを用いて「OpenFlow を用いた攻撃者遮断システム」を動作させ，性能評価をした．また，OpenFlow を用いた攻撃者遮断システムの遮断効果，遮断時の他の通信への影響について考察する．さらに，ACL を用いた従来の遮断手法を上記で示した環境，検証方法で動作させた．そして，OpenFlow を用いた攻撃者遮断手法での検証結果と比較し，有効性について考察する．

第 2 章では，不正通信検知システムの概要や攻撃者の従来の遮断手法について述べる．第 3 章では，OpenFlow を用いた攻撃者遮断手法の詳細，従来手法と提案手法の比較について述べる．第 4 章では，OpenFlow を用いた攻撃者遮断手法について評価する．第 5 章でまとめと今後の課題について述べる．

2. 不正通信検知システム

2.1 システムの概要

我々が開発・運用している不正通信検知システム [2][3] はインターネットから学内ネットワーク，また学内ネットワークからインターネットへ送信される TCP パケットのフラグやコネクションの接続状態に注目して不正通信を検知する．

2.2 攻撃者の遮断手法

本システムにおける攻撃者の遮断手法は以下の 2 つの方法がある．以下では，従来の 2 つの遮断手法についての特徴を述べる．

2.2.1 ACL を用いた遮断手法

ACL(Access Control List) は，LAN スイッチのインタフェースに適用することにより，そのインタフェース上で通過するパケットを許可したり，拒否することができる．通信アクセスの許可，拒否の基準としての宛先・送信元 MAC アドレス，宛先・送信元 IP アドレス，宛先・送信元ポート番号が利用できる．不正通信検知システムで用いた LAN スイッチでは 1 つのインタフェースに登録可能な攻撃者件数は最大 255 である．図 1 に ACL を用いた際の攻撃者の通信遮断手順を示す．

(1) 不正通信検知システムが攻撃者を検知.
(2) 攻撃者の送信元 IP アドレスを LAN スイッチの ACL に登録.
(3) 攻撃者がパケットを送信.
(4) 攻撃者のパケットは ACL を適用している LAN スイッチで破棄.

2.2.2 経路制御を用いた遮断手法

経路制御を用いた遮断手法では，学内ホストから攻撃者 IP アドレスへ送信される応答パケットを学内の特定のホスト (以下 NULL ホスト) 宛への静的経路を登録する．さらに，OSPF により経路情報を学内の各ルータへ配布し，攻撃者との通信を遮断する．図 2 に経路表を用いた際の攻撃者の通信遮断手順を示す．

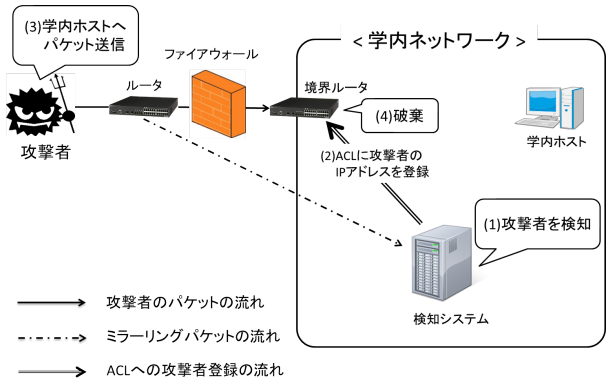

図 1 ACL を用いた攻撃者の通信遮断手順

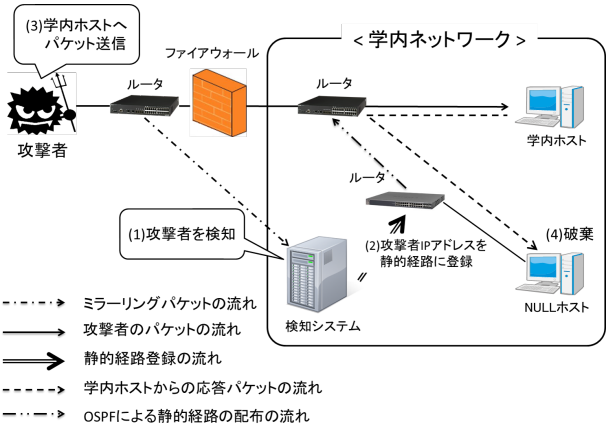

図 2 経路表を用いた攻撃者の通信遮断手順

(1) 不正通信検知システムが攻撃者を検知.
(2) 学内ネットワークの L3 スイッチへ攻撃者 IP アドレスの next hop を NULL ホストとする静的経路を登録．登録した経路情報は OSPF により学内の各ルータへ配布.
(3) 攻撃者がパケットを送信.
(4) 学内ホストからの応答パケットはすべて NULL ホストへ転送.

2.2.3 従来の遮断手法の問題点

2 つの攻撃者遮断手法にはそれぞれ問題点がある．ACL を用いた遮断手法では，LAN スイッチのソフトウェアによるが，1 つのインタフェースの ACL に登録可能な IP アドレス数は最大 255 件である．2014 年 1 月 1 日から 2014 年 5 月 1 日の不正通信検知システムが保持していた攻撃者数を調査したところ，最大攻撃者数は 610 件であり，上記の登録数では対応できない．さらに，攻撃パケットを遮断するには，ACL を適用した LAN スイッチを通過しないといけないため，すべての攻撃パケットを遮断するには，インターネットと学内ネットワークの境界に設置する必要がある．また，ACL の設定に関して，ACL はスイッチの各

表 1　マッチングルールで指定できる 12 種類の条件

Ingress Port	スイッチの物理ポート番号
Ether src	送信元 MAC アドレス
Ether dst	宛先 MAC アドレス
Ether type	Ethernet タイプ
IP src	送信元 IP アドレス
IP dst	宛先 IP アドレス
IP proto	IP プロトコル
IP ToS bits	IP の ToS 情報
TCP/UDP src port	TCP/UDP の送信元ポート番号
TCP/UDP dst port	TCP/UDP の宛先ポート番号
VLAN id	VLAN ID
VLAN priority	VLAN 優先度

ベンダごとに設定方法が異なるため，各ベンダの設定方法に合わせたプログラムを作成しなければならない．

経路制御を用いた遮断手法では，攻撃パケットに対する応答パケットを遮断することから，攻撃者のパケットが学内を通過してしまう．例えば，DoS 攻撃や DDoS 攻撃が学内の Web サーバを対象に行われたときに，攻撃者に対する応答パケットを遮断するため，Web サーバは攻撃者の通信を受信し続ける．そのため，Web サーバの処理性能が落ち，最悪の場合，サーバダウンする可能性がある．

3. OpenFlow を用いた攻撃者遮断手法

本章では不正通信検知システムと OpenFlow コントローラを連携させることで，動的なフローエントリを設定し，攻撃者の通信を遮断する手法について述べる．

3.1 OpenFlow

OpenFlow[5] は従来のスイッチ機能をデータプレーンとコントロールプレーンに分け，柔軟かつ集約的に通信を制御できる．コントロールプレーンは経路計算や受信パケットの扱い方を指示し，複数の OpenFlow スイッチを一元管理する．データプレーンは OpenFlow コントローラから設定されたフローエントリに基づき，パケットの転送や破棄などをする．また，フローエントリはマッチングルール・アクション・統計情報の 3 要素から構成されている．マッチングルールとして指定できる条件は，レイヤ 1 からレイヤ 4 までの情報であり，OpenFlow version1.0 では 12 種類の情報を扱うことができる．表 1 に OpenFlow version1.0 のマッチングルールで指定できる 12 種類の条件を示す．

3.2 攻撃者遮断手法

OpenFlow を用いた攻撃者遮断手法の構成図を図 3 に示す．本手法では，OpenFlow version1.0 を前提とする．

OpenFlow を用いた攻撃者遮断手法の概要を以下に示す．不正通信検知システムが攻撃者を検知すると，OpenFlow コントローラへ攻撃者 IP アドレスが通知される．また，攻撃者を遮断するフローエントリが OpenFlow スイッチへ書き込まれる．

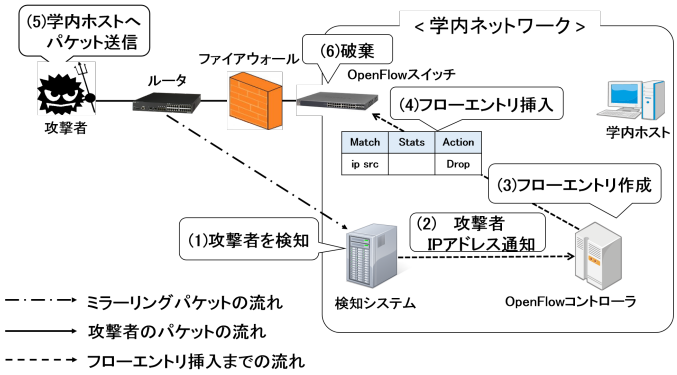

図 3　提案手法の構成図

図 3 に OpenFlow を用いた攻撃者の通信遮断手順について示す．

(1) 不正通信検知システムが攻撃者を検知．
(2) 不正通信検知システムは OpenFlow コントローラへ攻撃者 IP アドレスを通知．
(3) OpenFlow コントローラは通知された攻撃者 IP アドレスをもとにフローエントリを作成．
(4) OpenFlow コントローラは OpenFlow スイッチに対して FlowMod メッセージにより，フローエントリを書き込む．
(5) 攻撃者がパケットを送信．
(6) 攻撃者のパケットは OpenFlow スイッチでフローエントリのアクションリストの処理に従い，破棄される．

3.3 従来手法との比較

3.3.1 ACL を用いた攻撃者遮断手法との比較

ACL を用いた攻撃者遮断手法で使用しているスイッチの ACL の最大登録件数は 255 件である．2.2 節で示した不正通信検知システムが保持していた最大攻撃者数が 610 件であったことから，ACL では対応できない．一方，OpenFlow 攻撃者遮断手法で用いた NEC 社製の OpenFlow スイッチ UNIVERGE PF5220[6] の最大フローエントリ数は 80000 エントリであるため，上記の攻撃者数に対応できると考えられる．また，ACL の設定方法は各ベンダによって異なるが，OpenFlow では公開された OpenFlow プロトコルに基づいて，各ベンダはスイッチを設計，実装するため，OpenFlow コントローラからあらゆるベンダー製の OpenFlow スイッチを OpenFlow という統一した手法により制御できる．

3.3.2 経路制御を用いた攻撃者遮断手法との比較

経路制御を用いた攻撃者遮断手法では，攻撃パケットに対する応答パケットを遮断することから，攻撃者のパケットが学内を通過してしまう問題点がある．OpenFlow を用いると，攻撃者の通信を遮断するフローエントリが OpenFlow スイッチへ設定された後は，標的ホストに攻撃

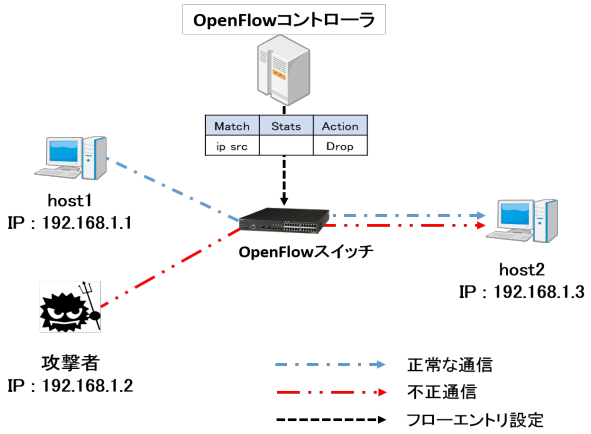

図 4　実験環境図

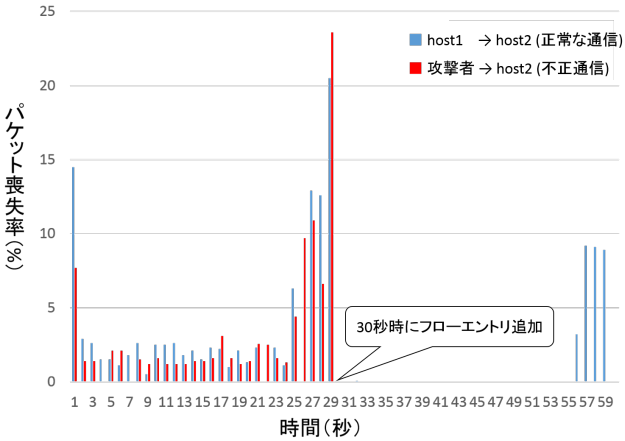

図 5　1 秒毎のパケット喪失率

```
09:47:29.920368 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
09:47:29.920434 IP 192.168.1.2.56086 > 192.168.1.3.3000: UDP, length 1470
09:47:29.920438 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
09:47:29.920477 IP 192.168.1.2.56086 > 192.168.1.3.3000: UDP, length 1470
09:47:29.920513 IP 192.168.1.2.56086 > 192.168.1.3.3000: UDP, length 1470
09:47:29.920794 IP 192.168.1.2.56086 > 192.168.1.3.3000: UDP, length 1470
09:47:29.920799 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
09:47:29.920840 IP 192.168.1.2.56086 > 192.168.1.3.3000: UDP, length 1470
09:47:30.011895 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
09:47:30.012114 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
09:47:30.012304 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
09:47:30.012415 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
09:47:30.012645 IP 192.168.1.1.55619 > 192.168.1.3.4000: UDP, length 1470
```

図 6　host2 の攻撃者遮断時でのパケット受信ログ

者のパケットは届くことなく OpenFlow スイッチで破棄することが可能である．

3.4　仮想環境における OpenFlow を用いた攻撃者遮断手法の検証

我々は，これまでに OpenFlow を用いた攻撃者遮断システムの有用性を検証するために，Open vSwitch version1.10 を用いて仮想環境での実験を実施してきた．以下に仮想環境での動作検証について述べる．

3.4.1　実験概要

OpenFlow を用いた攻撃者遮断手法が，攻撃者の通信を遮断できることを検証する．本実験では，iperf[7] を用いて正常な通信と不正な通信を再現する．攻撃者の通信を遮断するフローエントリを OpenFlow コントローラから OpenFlow スイッチへ設定することにより，遮断動作が正常に行えているか検証する．実験環境を図 4 に示す．実験の手順を以下に示す．

- host1 から host2 に対して，iperf により，帯域幅 50Mbps でパケットを 60 秒間送出し続ける．
- 攻撃者から host2 に対して，iperf により，帯域幅 100Mbps でパケットを 60 秒間送出し続ける．
- 実験開始後 30 秒時に OpenFlow コントローラから OpenFlow スイッチに対して攻撃者の遮断をするフローエントリを設定する．

3.4.2　実験結果と考察

実験 10 回分の 1 秒間の正常な通信と不正な通信の喪失率の平均を図 5 に示す．また，host2 での攻撃遮断時の tcpdump で収集したログを図 6 に示す．

図 5 より，攻撃者遮断前と攻撃者遮断後の正常な通信の喪失率を見ると，攻撃者遮断後は 0%まで下がっており，攻撃者遮断後の通信は安定していることがわかる．また，図 6 より，攻撃者遮断後 (30 秒後) は，攻撃者 (IP アドレス：192.168.1.2) からのパケットは途絶えている．これにより，攻撃者の通信を遮断できていることが確認できる．

仮想環境での実験において，OpenFlow コントローラから OpenFlow スイッチに対して，送信元 IP アドレスを識別して破棄するフローエントリを書き込み，攻撃者の遮断ができていることを確認できた．

4.　性能評価実験

本節では，OpenFlow を用いた攻撃者遮断システムの性能評価実験の環境，検証方法，また，検証結果について述べる．

4.1　実験概要

学内ネットワーク内の Web サーバに学外ネットワークに存在する攻撃者から 80 番ポートに DoS 攻撃を受けることを想定して，実験環境を構築した．環境として，trema version0.4.6[8] で作成した OpenFlow コントローラ 1 台，NEC 社製の OpenFlow スイッチ UNIVERGE PF5220 を 1 台，Web サーバを 1 台，実験 1 ではホスト 4 台，実験 2 ではホスト 7 台を用いた．

```
# config
# ip access-list standard test
# 1 deny [攻撃者 IP アドレス]
# exit
```

図 7　ACL への攻撃者登録例

表 2　攻撃者の通信を遮断するフローエントリ

マッチングルール	統計情報	アクション
src_IP : [攻撃者 IP アドレス]	—	drop

4.2　攻撃の遮断

不正通信検知システムの攻撃者検知基準は，1 秒に 10 コネクションを越えるホストは攻撃者とみなされる．DoS 攻撃などの短時間で大量のコネクション要求を送る攻撃は検知基準により，1 秒で検知されることになる．本実験では，攻撃者の通信の遮断による他の通信への影響や遮断命令時のパケットロスの割合を調査するため，攻撃を受けてから 10 秒経過後，攻撃者を遮断した．

遮断方法に関して，ACL を用いる手法では，不正通信検知システムから攻撃者 IP アドレスを取得後，telnet を用いてスイッチの ACL に遮断命令を送信することにより攻撃者の通信を遮断する．UNIVERGE PF5220 の ACL 登録例を図 7 に示す．UNIVERGE PF5220 の ACL の 1 つのインターフェースに適用可能な登録件数は，最大 511 件である．

OpenFlow では，不正通信検知システムから攻撃者 IP アドレスを OpenFlow コントローラが受け取る．その後，攻撃者 IP アドレスをマッチングルール，アクションを Drop としたフローエントリを作成し，OpenFlow スイッチへ設定することで攻撃者の通信を遮断する．攻撃者の通信を遮断するフローエントリを表 2 に示す．

4.3　検証方法

攻撃者から Web サーバに対して DoS 攻撃があった場合，攻撃者を遮断する際に，Web サーバと Web クライアントの通信に影響がないか検証する．評価の指標として Web サーバからの応答率，応答時間を用いる．

4.4　実験 1 の概要

実験 1 では，従来の ACL を用いた攻撃者遮断手法と今回提案する OpenFlow を用いた攻撃者遮断手法で同様の実験を行い，それぞれの手法の有効性を検証する．ACL を用いた攻撃者遮断手法での実験環境を図 8，アクセスリストを表 3 に，OpenFlow を用いた攻撃者遮断手法での実験環境を図 9，フローテーブルを表 4，タイムチャートを図 10 に示す．

表 4 のフローエントリ 1〜5 は正常な通信を行うホスト

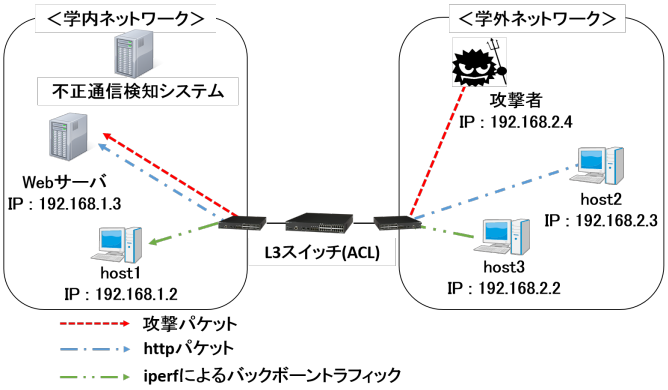

図 8　ACL 遮断手法を用いた実験環境

表 3　実験で設定したアクセスリスト

適用順序	アクセスリスト
1	deny 192.168.2.4
1000	permited any

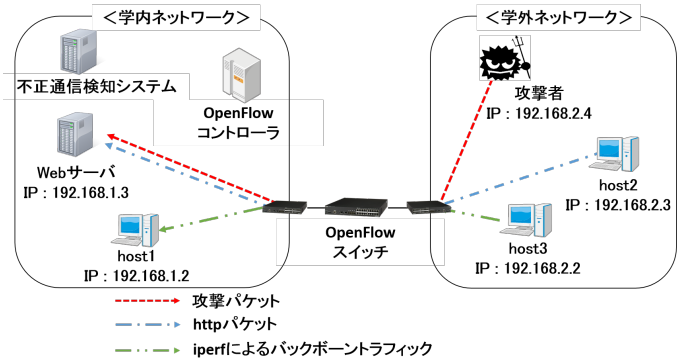

図 9　OpenFlow 遮断手法を用いた実験環境

表 4　フローテーブル

フローエントリ	マッチングルール	統計情報	アクション
1	in_port=3, src_MAC=00:90:f5:c9:8e:46, dst_MAC=00:12:e2:00:00:0a, src_IP=192.168.2.4, dst_IP=192.168.1.3	—	action1 : set_dl_src = 00:12:e2:00:00:64 action2 : set_dl_dst = 00:26:b9:12:67:57 action3 : output
2	in_port=2, src_MAC=00:26:b9:12:67:57, dst_MAC=00:12:e2:00:00:64, src_IP=192.168.1.3, dst_IP=192.168.2.4	—	action1 : set_dl_src = 00:12:e2:00:00:0a action2 : set_dl_dst = 00:90:f5:c9:8e:46 action3 : output
3	in_port=3, src_MAC=00:20:30:f5:20:cf, dst_MAC=00:12:e2:00:00:0a, src_IP=192.168.2.3, dst_IP=192.168.1.3	—	action1 : set_dl_src = 00:12:e2:00:00:64 action2 : set_dl_dst = 00:26:b9:12:67:57 action3 : output
4	in_port=2, src_MAC=00:26:b9:12:67:57, dst_MAC=00:12:e2:00:00:64, src_IP=192.168.1.3, dst_IP=192.168.2.3	—	action1 : set_dl_src = 00:12:e2:00:00:0a action2 : set_dl_dst = 00:20:30:f5:20:cf action3 : output
5	in_port=3, src_MAC=00:12:33:b3:27:46, dst_MAC=00:12:e2:00:00:0a, src_IP=192.168.2.2, dst_IP=192.168.1.2	—	action1 : set_dl_src = 00:12:e2:00:00:64 action2 : set_dl_dst = 00:2b:13:4b:10:0f action3 : output
6	src_IP=192.168.2.4	—	drop

間の通信を処理するフローエントリである．フローエントリ 6 は，攻撃者の通信を遮断するフローエントリである．実験方法として図 10 より，以下に示す方法で実験をする．

- host2 から Web サーバに対して，60 秒間 httperf[9] により http リクエストを 200req/sec の割合で送信する．
- host3 から host1 に，バックボーントラフィックとして iperf により，帯域幅 100Mbps でパケットを 60 秒

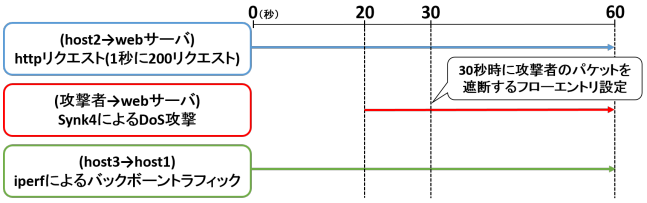

図 10　タイムチャート図

表 5　実験 1：Web サーバからの応答率 (ACL を用いた場合)

	0~10 秒	10~20 秒	20~30 秒	30~40 秒	40~50 秒	50~60 秒
応答率 (%)	100	100	96.98	100	100	100

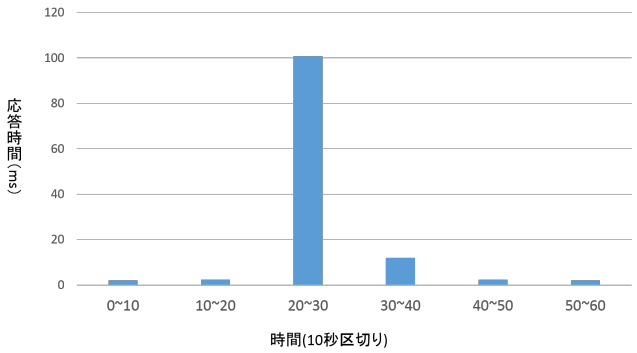

図 11　実験 1：Web サーバからの応答時間 (ACL を用いた場合)

表 6　実験 1：Web サーバからの応答率 (OpenFlow を用いた場合)

	0~10 秒	10~20 秒	20~30 秒	30~40 秒	40~50 秒	50~60 秒
応答率 (%)	100	100	98.65	100	100	100

間送出し続ける．

- 攻撃者は Web サーバの 80 番ポートに対して，実験開始後 20 秒時に DoS 攻撃を 1 秒間に 50000 パケットの割合で仕掛ける．
- 実験開始後 30 秒時に攻撃者を検知したとみなし，攻撃者の遮断をする．

4.5　実験結果と考察

ACL を用いた攻撃者遮断手法の実験結果として，実験 10 回分，10 秒ごとの Web サーバの応答率を表 5 に示す．また，応答時間の平均を図 11 に示す．OpenFlow を用いた攻撃者遮断手法の実験結果として，実験 10 回分，10 秒ごとの Web サーバの応答率を表 6 に示す．また，応答時間の平均を図 12 に示す．

ACL を用いた攻撃者遮断手法では，表 5 から，DoS 攻撃遮断後 (30 秒後) からの応答率は 100%に戻っている．また，図 11 より，DoS 攻撃遮断後 (30 秒後) からの応答時間は約 12ms まで下がっており，平常時の平均値である 2.2ms に近づいている．これらの結果から，ACL を用いた攻撃者の遮断手法は，攻撃者を遮断することにより他の通信は安定して行えている．

OpenFlow を用いた攻撃者遮断手法では，表 6 より，DoS

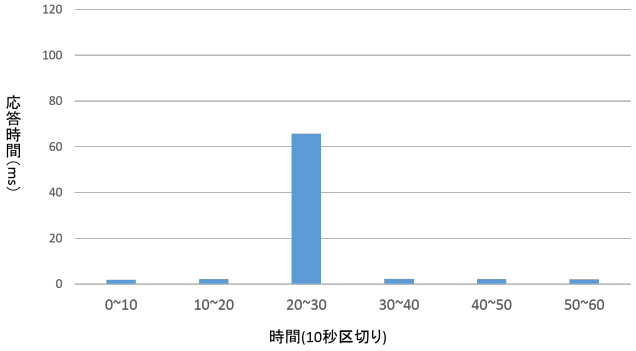

図 12　実験 1：Web サーバからの応答時間 (OpenFlow を用いた場合)

表 7　実験 2：Web サーバからの応答率 (ACL を用いた場合)

	0~10 秒	10~20 秒	20~30 秒	30~40 秒	40~50 秒	50~60 秒
応答率 (%)	100	100	99.12	99.98	100	100

攻撃遮断後 (30 秒後) からの応答率は 100%に戻っている．また，図 12 より，DoS 攻撃遮断後 (30 秒後) からの応答時間は約 2.29ms まで下がっており，平常時の平均値である 2.0ms に近づいている．これらの結果より，OpenFlow を用いた攻撃者の遮断手法は，攻撃者を遮断することにより他の通信は安定して行えている．

2 つの手法の実験結果から，OpenFlow を用いた攻撃者遮断手法，ACL を用いた攻撃者遮断手法は両者とも攻撃遮断後の通信の傾向から，他の通信に影響を及ぼすことなく遮断できている．

4.6　実験 2 の概要

実験 2 では，それぞれの遮断手法の遮断命令書き込み数に対応するパケットロスの割合について検証する．実験環境は，実験 1 の環境構成に攻撃者を 3 台増やした環境とする．

実験の流れとして，実験 1 と同様に図 10 のタイムチャートに従う．

4.7　実験結果と考察

ACL を用いた攻撃者遮断手法の実験結果として，実験 10 回分，10 秒ごとの Web サーバの応答率を表 7 に示す．また，応答時間の平均を図 13 に示す．OpenFlow を用いた攻撃者遮断手法の実験結果として，実験 10 回分，10 秒ごとの Web サーバの応答率を表 8 に示す．また，応答時間の平均を図 14 に示す．

ACL を用いた攻撃者遮断手法では，表 7 より，DDoS 攻撃遮断後 (30 秒後) からの応答率は 99.98%であり，平常時より少し低下している．また，図 13 より，DDoS 攻撃遮断後 (30 秒後) からの応答時間は，約 103ms であり，平常時の 2.31ms と比べると長い．ACL では telnet を用いてスイッチの ACL に対して遮断する命令を送信することで攻

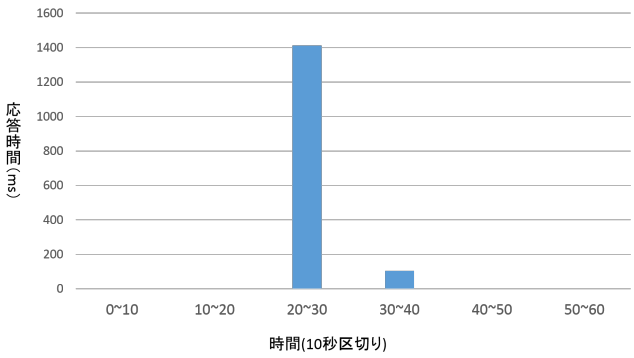

図 13　実験 2：Web サーバからの応答時間 (ACL を用いた場合)

表 8　実験 2：Web サーバからの応答率 (OpenFlow を用いた場合)

	0~10 秒	10~20 秒	20~30 秒	30~40 秒	40~50 秒	50~60 秒
応答率 (%)	100	100	99.13	100	100	100

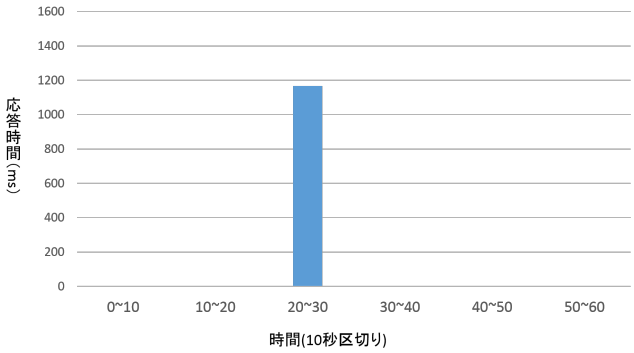

図 14　実験 2：Web サーバからの応答時間 (OpenFlow を用いた場合)

撃者を登録する．その際に，パスワード入力などの手間がかかるため，タイムロスが生じる．そのため，攻撃者を登録するまでの過程の間に攻撃者のパケットを通してしまい，上記の結果が得られたと考えられる．

OpenFlow を用いた攻撃者遮断手法では，表 8 より，DDoS 攻撃遮断後 (30 秒後) からの応答率は 100%に戻っている．また，図 13 より，DDoS 攻撃遮断後 (30 秒後) からの応答時間は，約 2.56ms であり，平常時の挙動に戻っていることがわかる．これらの結果より，OpenFlow を用いた攻撃者の遮断手法は，複数の攻撃者の登録をする状況においても攻撃者の遮断により，他の通信には影響を及ぼさないことがわかった．

5. おわりに

5.1 まとめ

本論文では，ACL，OpenFlow を用いた 2 つの手法に対して実験を行い，比較することにより OpenFlow を用いた攻撃者遮断手法の有用性について考察した．実験 1 より，両手法とも攻撃遮断後の通信の傾向は，Web サーバの応答率と応答時間がともに平常時に近い挙動に戻っていたことが分かった．攻撃者の通信の遮断により，他の通信に影響を及ぼさないという観点から，遮断効果が得られていると考えられる．実験 2 では，両者の遮断手法の遮断命令の書き込み数に対応するパケットロスの割合について検証した．ACL による攻撃者の登録は，telnet を用いてスイッチにログインし，攻撃者を遮断する命令を送信することで攻撃者の通信を遮断する．そのため，スイッチの ACL に対して遮断命令をしている間に攻撃者のパケットが通過してしまう．一方，OpenFlow では，OpenFlow コントローラで生成したフローエントリを OpenFlow スイッチに対して，FlowMod メッセージを送るだけで，フローエントリの登録，つまり，攻撃者の登録ができる．そのため，ACL を用いた手法よりも遮断命令の書き込み数に対応するパケットロスは少ない．また，ACL はスイッチの各ベンダごとに設定方法が異なるため，各ベンダの設定方法に合わせたプログラムを作成しなければならない．しかし，OpenFlow では公開された OpenFlow プロトコルに基づいて，各ベンダはスイッチを設計，実装するため，OpenFlow コントローラからあらゆるベンダ製の OpenFlow スイッチを OpenFlow という統一した手法により制御ができる．

これらのことから，OpenFlow を用いた攻撃者遮断手法は有効であると考えられる．

5.2 今後の課題

本論文では，2 つの実験をもとに検証を行ったが，実環境で運営するためにはさらに検証が必要である．OpenFlow スイッチでのフローエントリが増えた場合の転送性能，OpenFlow スイッチが動作できる最大フロー数，OpenFlow コントローラ，OpenFlow スイッチ間の接続性を調査する必要がある．また，経路制御を用いた攻撃者遮断手法と比較する必要がある．

参考文献

[1] 警察庁：インターネット観測結果等 (平成 25 年度第 2/四半期 (7 月～9 月)),http://www/npa.go.jp/cyberpolice/detect/pdf/20131024.pdf，2013 年 12 月
[2] 有馬竜昭，熊谷悠平，永山聖希，吉田和幸：scan 攻撃の検知とその遮断について. 情報処理学会マルチメディア，分散, 協調とモバイル (DICOMO2007) シンポジウム論文集,pp. 1748-1753, 2007 年 7 月
[3] 小刀祢知哉，天本大地，小埜勇貴，有馬竜昭，池部実，吉田和幸：scan 攻撃検知システムを用いた被検知ホストの挙動についての調査. 第 65 回電気関係学会九州支部連合大会, pp. 1-1, 2012 年 9 月
[4] 下川大貴，小刀祢知哉，池部実，吉田和幸：OpenFlow を用いた攻撃者の遮断方法に関する一考察 第 66 回電気関係学会九州支部連合大会, pp. 1-1, 2013 年 9 月
[5] Open Networking Foundation，http://www.opennetworking.org/
[6] UNIVERGE PF5220，http://jpn.nec.com/univerge/pflow/pfs.html#pf5220
[7] iperf http://iperf.fr/
[8] TREMA - An OpenFlow Controller，http://www.trema.info

[9] httperf http://www.hpl.hp.com/research/linux/httperf