論文　**Original Paper**

OpenFlowを用いた小規模会議向け ネットワーク個人認証システムの実装

森　重　駿　一*，中　村　嘉　志*

A Study on an Implementation of Personal Identification System for Small Meetings using OpenFlow Virtual Networks

Shunichi Morishige*，Yoshiyuki Nakamura*

概要: 現在，あらゆる場所でインターネットを使うことができる。特に会議などにおいて，調査や補助の目的でインターネットを利用したいという要求も多い。会議室のような場所でこの要求を満たそうとすると，計算機ネットワーク環境の臨時設営やその上でのセキュリティ対策が問題となる。一般に会議では不特定の利用者が計算機ネットワークに接続するため，ひとたびトラブルが発生した際にはどの利用者が問題を発生させているか分かりづらく，原因の究明が難しいからである。そこで本稿では，これらの問題をOpenFlowというネットワークの仮想化技術を用いて解決することを試みる。特に，臨時に展開する計算機ネットワーク環境において，その困難さと課題を明らかにし，それらを改善する目的で個人認証を簡便に行う仕組みの検討および実現を行う。

Key words: 計算機ネットワーク，臨時設営ネットワーク，SDN，Web認証，CGI

1. はじめに

現在，インターネットはあらゆる場所で利用可能である。会議などの場面においても，調査や補助の目的で使用したいという要求も多い。そのため，インターネットへ接続するための計算機ネットワーク環境（以下，ネットワークは，計算機ネットワークを表すものとする）が整備されている。

一般的なネットワーク環境は，大きく2つに分類することができる。会社や家庭内など特定の人物が使用する環境と，それとは対照的に公共施設や臨時のイベント会場など不特定の人物が使用する環境である。このうち，特定の人物が使用するネットワーク環境では，接続している人の管理を行うことは容易である。しかし，不特定の人物が使用するネットワーク環境では，セキュリティ面での問題が発生する可能性がある。例えば端末の乗っ取りや盗聴，ネットワーク資源の独占などが挙げられる。特に無線LANを用いた場合，通信用の電波を傍受することが容易であるため，これらの問題はより顕著となる。ホテルや空港で無料で利用できるネットワークでは，実問題が発生している[1)]。ネットワークを安全に使用するためにも，ネットワークの管理者は誰がどのような目的で使用しているかを確認する必要がある。

不特定の利用者が使用するネットワーク環境についての研究，実例はこれまで数多く存在する[2−4)]。しかし，不特定の人物が使用する臨時の小規模なネットワークについてはあまり議論がされていない。なぜなら，万人に共通した場面想定が難しく，実際に恩恵を受ける人が少ないからである。

ここで，臨時にネットワーク環境を構築することを考える。一般に，小会議室で行われる小規模な会議では20人程度が定員である[5)]。そこで，20人が参加する会議を想定すると，これらの設営はネットワークの構築経験が浅い人には難しい。ネットワーク設備自体の費用が高く，設営に専門知識が必要となるからである。設営を外部に委託することも可能ではあるが，委託費用などが発生してしまう。小規模の会議において，これらの費用を参加者で負担することは現実的ではない。

このように，ネットワークの構築経験が浅い人が小規模の会議に合わせたインターネット接続環境を臨時に設営することは簡単ではない。そこで，臨時にネットワークを構築し，参加者が持参する計算機をインターネット接続させることに重きを置き，管理者にとって簡便なシステムの構築を検討する。本稿では，従来のネットワーク環境の問題点を指摘し，仮想ネットワークである

* 国士舘大学理工学部
School of Science and Engineering, Kokushikan University

OpenFlowを用いてそれらの問題を解決する。

第2章では，不特定の人物が使用するインターネット環境についての問題点と関連研究について述べる。第3章では，本研究で用いるOpenFlowというネットワークを仮想化する技術について紹介する。第4章では，提案システムの設計方針，続く第5章では，その実装についてそれぞれ詳述する。第6章では，実験と考察によって提案システムを評価する。第7章は，まとめである。

2. 研究課題

2.1　既存のネットワーク設備の問題点

ネットワーク環境を構築するのみであれば，市販されている機器をそのまま使用すれば良い。しかし，不特定の人物が集う小規模な会議において，従来のネットワーク機器をそのまま使用するには問題が大きく2つ存在する。

1つ目は，費用が高額になってしまう点が挙げられる。市販されているネットワーク機器の価格帯を図1に示す。大規模向けのネットワーク機器は，大勢の人が同時に使うことを想定しており，機器の処理能力が高い。さらに，初めから認証の仕組みを搭載しているものも存在する。これらは会議において使用するには十分な能力を有するが，その費用を全員で負担することは難しい。一方，家庭用のネットワーク機器は安価である。さらに持ち運びやすく設営も簡単であるが，同時接続台数が限定されており処理能力が不足する。また，認証の仕組みが存在しないため不特定の人物が使用する環境で利用することは難しい。

これらの中間である10～100人程度を対象としたネットワーク設備はあまり存在しない。なぜならば費用対効果が薄く，メーカーが開発・発売を行わないからである。したがって，上記のような会議の環境においても，より大規模なシステムを使用せざるを得ず，費用が高くなってしまう。

2つ目は，ネットワークが不正利用された場合に，原因を追求しづらい点である。不特定の人物が使用するネットワークは，適切に扱われるとは限らない。端末の乗っ取りや盗聴，ネットワーク資源の独占など，管理者の意図しない使われ方をする場合がある。また，普通に使用している人ががネットワーク全体に負荷をかけている可能性も考えられる。これらへの対策として，接続端末のフィルタリングや検疫ネットワークの導入などが行なわれているが，原因の特定は難しい。ネットワークの管理者は，不正利用者を推測し，忠告や利用の停止を行う必要がある。そのためには，ネットワークに接続している端末と使用している人物を結びつける認証を行う必要が生じる。

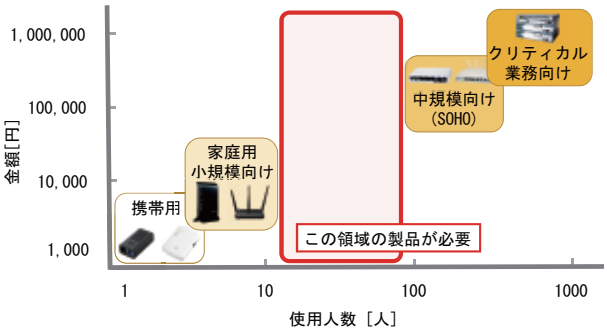

図1　ネットワーク機器の価格帯

2.2　関連研究

不特定の人物が使用するネットワーク環境に関する研究は多く行われている。関連するものを以下に示す。

後藤らは，eduroamと呼ばれる世界中で使われる学術系認証システムの導入，運用の報告をしている[2]。RADIUSと呼ばれるサーバーを階層的に用いた認証システムを使用し，遠隔地においても同じアカウントで認証を行うことができるものである。

櫻田らは，東京農工大学のインターネットを仮想化技術を用いて構築したことを報告している[3]。大学における従来の無線LANシステムの問題を洗い出し，新システムの設計と構築を行うものである。認証にはIEEE802.1X認証とWeb認証を用いている。

また中村らは，学会の会場においてインターネット接続環境を整備する方法論を示している[4]。200人程度が参加する会場において，ソフトウェアルーターを用いることで，その環境を構築・運用できるとしている。

文献[2,3]に代表されるように，建物に併設される固定型のネットワーク環境の構築に関する論文は多く存在する。しかし，文献[4]で提示されるように外出先などにおいて臨時に展開するネットワーク環境についてはあまり議論がされていない。また，対象とする使用人数も大勢である。そのため，本稿のターゲットである小規模な会議で使用する用途とは異なる。

文献[3]で触れられているが，管理者がネットワークを構築する負担は大きい。そのため，昨今のネットワーク環境には仮想化の技術が用いられる場合がある。仮想化には，低費用，集中管理，ソフトウェアによるネットワークの動的変更などの利点が存在する。この文献では，ネットワークの仮想化基盤を用いて柔軟なネットワークの構築を実現している。

一方松谷は，ARPのブロードキャストパケットを監視し，不正な通信を排除する仕組みを提案している[6]。これはRADIUSなどに対応していない既存の設備においても，パケットを監視することによって不正接続が把握可能であることを示した。

ネットワーク中のパケットには，送信元や宛先，送信するデータなどの情報が記載されている。これらの情報を用いることで，送信元である端末を把握することは可

能である。攻撃自体を防ぐことは難しいが，誰が行ったのかさえ把握することができれば，通信を排除することはできる。そこで，不正利用を行う人を特定する仕組みがあれば問題の解決が可能である。

2.3 実装方針

関連研究より共通して分かることは，不特定の人物が使用するネットワーク環境では，不正利用への対策を行う必要があるということである。その対策の1つに個人を特定する個人認証という考え方がある。

認証の仕組みは高価なネットワーク機器であれば予め実装されている。しかし，ネットワーク中の全ての機器が同じ規格に対応しなければならないため，費用が増大する原因となる。

そこで，本研究で取り上げた問題に対して，ネットワークの仮想化技術を使用し対処することとする。ネットワークの仮想化を用いることで，動的な変更を行うことが可能となる。そのため，ネットワークの構築と個人の識別を実現することができると考える。ネットワーク仮想化の仕組みとして，OpenFlowを用いるものとする。

3. OpenFlow

3.1 OpenFlowとは

ネットワークの仮想化を行う考え方に，SDN（Software Defined Network）というものがある。これはソフトウェアで定義されたネットワークという意味で，ネットワーク全体を仮想化してソフトウェアによって構築することを指す。実装の一つにOpenFlow[7]がある。

OpenFlowは，元々大学において自由な構築を行うための実験用のネットワークとして発案されたものである。従来のネットワーク内に仮想的に通信用と制御用のネットワークを作成し，ソフトウェアによるネットワーク環境の構築を実現している。データ通信は従来のネットワークと同一の仕組みで行うため互換性があり，末端に接続することも容易である。OpenFlowの特徴は文献[8]に詳しい。

OpenFlowの仕組みはスイッチとコントローラの大きく2つに分けられる。各スイッチにはフローテーブルと呼ばれるパケットの条件と処理方法が示された表が存在する。フローテーブルは，パケットの処理条件と動作が記載されているレコードで構成されている。スイッチに到着したパケットはフローテーブルの条件に照らし合わせられ，条件に合致した場合はレコードに対応する動作を行う仕組みである。

パケットがフローテーブルの条件に一致しない場合，スイッチはコントローラに問い合わせを行う。この時，パケットはコントローラで実装されたプログラムにしたがって処理される。また，コントローラはフローテーブルの内容を書き換えることや，パケットの通信ログを参照することが可能である。そのため，ソフトウェアでのネットワーク構築が可能である。OpenFlowの概略図を図2に示す。

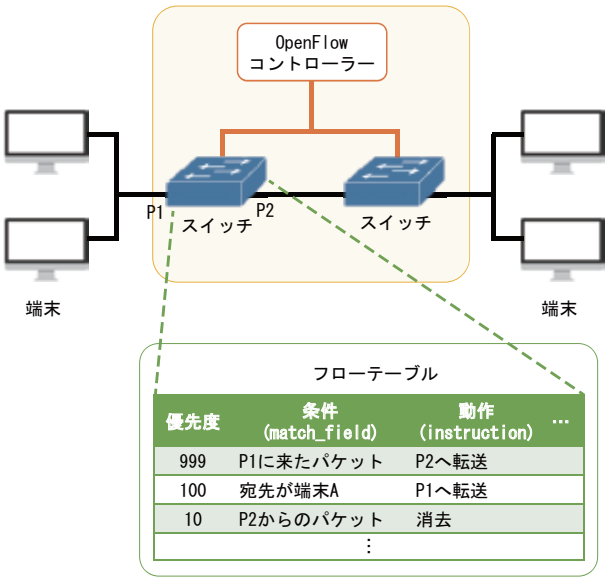

図2 OpenFlowの概略図とフローテーブル

3.2 OpenFlowの利点と欠点

OpenFlowを用いる利点は，ネットワークを柔軟に管理できることである。OpenFlowネットワーク中のすべての情報はコントローラに集められることとなる。一方，フローテーブルに該当するレコードがあるパケットは，スイッチ上で転送処理が行われる。つまり，ネットワーク全体を管理しながら各スイッチに処理を分散することができる。そのため，認証システムがすべてのパケットを確認する必要がなくなり，認証待ちなどのネットワーク遅延による応答待ちの原因を取り除くことができる。

OpenFlowの欠点はソフトウェア実装によるレイテンシ（通信の遅延時間）の増大である。ネットワークシステムの性能は，仮想化を実現しているとプログラムの性能により左右される。しかし，利点でも述べたようにパケットの判定はスイッチ上で行われるため，この欠点を補うことができると考えられる。

以上より，認証システムが必要なネットワークに，OpenFlowを導入することが最適であると考えた。本稿では，OpenFlowを用いたネットワークについて個人を識別することについて議論する。

4. システム設計

本システムは，臨時にネットワークを構築したい場合に使用するものである。小規模な会議を想定し，20人程度が使用するシステムとする。管理者が各個人を特定できるような，安全なネットワークの構築を補助する仕組みを実現する。

4.1　個人の特定方法

本研究では既存のネットワーク機器を臨時に使用する時の問題点として，費用やネットワークの知識の必要性，不正利用対策を挙げた。これらの問題点を解決するために，個人の特定を行う必要がある。そのために認証システムを実現する。認証とは，端末と利用者を結びつけ，ネットワークの使用許可を与えるかどうかを判断するものである。**表1** に，ネットワークの認証システムについて代表的なものをまとめる。このうち，IEEE802.1X認証は多くの実装例が存在するが，初回認証が煩雑である。一方Web認証は，CGIにより独自の実装が可能である。

本システムではWeb認証を用いることとする。今回使用するOpenFlowでは，パケットに対する処理をスイッチ上のフローテーブルに記録することが可能である。認証可否の情報を直接フローテーブルに書き込むため，独自の仕組みを構築できるWeb認証が最適であると考える。したがって，本研究ではこの認証システムを導入する。

また認証には利用者の情報と端末の情報が必要となる。本システムでは，利用者情報として事前にユーザIDとパスワードを配布する形式を取ることとした。また初回接続時に，端末の情報として接続したポート情報，IPアドレス，MACアドレスを取得する。そのため，端末情報から利用者を特定することが可能であると考える。この仕組みにより，もしネットワークの不正利用や資源の独占が発生した場合，使用者のパケットをスイッチ上で消去し，擬似的に接続を拒否することができる。

4.2　認証の手順

本システムでの認証の手順は以下の通りである。(1) ネットワークの使用者に，予めユーザーIDとパスワードを配布する。(2) 使用者がネットワークに接続した際，未認証の端末から送信されたHTTP（S）のパケットに対して，コントローラ上で宛先を自身の認証サーバに書き換える。(3) 強制的にログイン画面が表示され，使用者はユーザーIDとパスワードを入力する。(4) この時認証システムは，端末のIPアドレス，MACアドレス，接続したポート番号を取得する。(5) 認証システムは使用者が入力したユーザー名とパスワードを，システムのテーブルに存在するものと比較する。(6) 合致していたら，認証情報をフローテーブルに書き込み，インターネットへ接続可能にする。このような手順でネットワークへの接続を許可する。

本来，認証システムは，すべてのパケットに対して判定を行うべきである。しかしARPやpingなどのネットワークの制御，管理のための通信（ICMPパケット）を制限すると，通信が拒否されている場合にネットワークに接続することができず，実験に支障をきたす。そのため，今回はこれらの通信を制限しないこととする。

表1　代表的な認証システム

認証方式	内容	認証キー
MACアドレス認証	MACアドレスを使用 詐称される恐れがある	MACアドレス
IEEE802.1x認証	IEEE802.1xプロトコルを使用 認証にRADIUSサーバーを用いる	ユーザー名＋パスワード 証明書(SIMカードなど)
Web認証	HTTP(S)プロトコルを使用 認証をサーバーで独自実装する	ユーザー名＋パスワード など

4.3　コントローラ設計

本システムのコントローラの処理について示す。スイッチ上のフローテーブルの条件は，パケットが以下の3つの要素に合致する場合である。1つ目はパケットが到着したスイッチのIDとポート番号である。2つ目は送信元のIPアドレスである。3つ目は送信元のMACアドレスである。フローテーブルは，インターネットへの接続先となる次のスイッチ（Next Hop）にパケットを送信する。以上のコントローラの処理を**図3** に示す。

またこれらとは別に，一定時間おきにフローテーブルを監視するプログラムを稼働させる。認証済みである端末情報はフローテーブルに書き込まれるが，一度書き込まれたものは，自動的に消去されることはない。そのため，安全性を高めるために30分間通信が行われていない端末は認証を取り消すこととする。もし不正利用をされた場合には，管理者が特定のユーザを対象に拒否コマンドを実行することで，使用者のネットワーク利用を停止することができる。

4.4　既存のネットワークへの接続方法

本システムは，既存のネットワーク環境に割り込む形で設置し，利用者に認証システム付きのインターネットを提供するものである。その際，既存のネットワーク環境に接続するために環境設定を行う必要がある。具体的には，各ポートのMACアドレス，IPアドレスとネットマスク，Next Hopの設定が必要である。今回，ネット

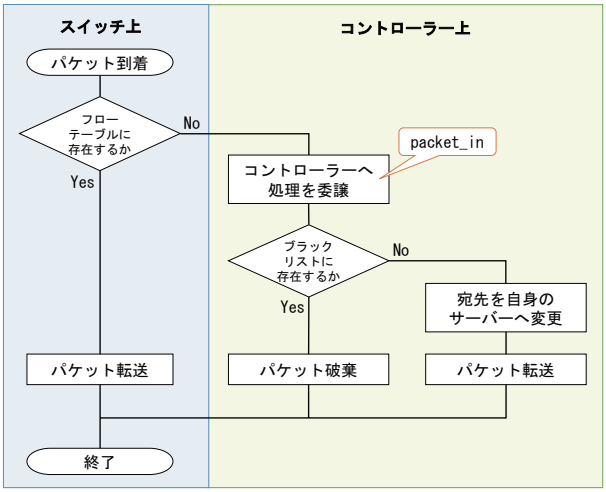

図3　パケット処理のフロチャート

表2　使用したソフトウェア

項目	ソフトウェア
マシン	Raspberry Pi 3 Model B
OS	Ubuntu MATE 16.04.1 LTS
開発言語	Ruby 2.3.1p112
パッケージ管理	RubyGems 2.5.1
仮想スイッチ	Open vSwitch 2.5.0
OpenFlow	trema version 0.10.0
Webサーバー	Apache HTTP Server 2.4.6

ワークの環境設定はテキストファイルの書き込みにより行う方式とした。管理者，すなわち会議において参加者にインターネット接続を提供する者は，設定が終わり次第，既存のネットワークに本システムを接続し，環境を構築する。

5. システム実装

本システムでは，実装端末としてRaspberry Pi 3，オペレーティングシステムにUbuntu MATE，コントローラにTrema[9]，仮想スイッチの構築にOpen vSwitch[10]を用いる。開発言語はRubyを用いる。使用したソフトウェアを表2に示す。

コントローラにTremaを用いた理由は3つある。1つ目は，オープンソースのプログラムであるため動作を理解し，独自の処理を実装することが容易であるからである。開発元がIT企業のNECであるため文献や資料が多く，フレームワークの完成度も評価できる。2つ目は，コントローラの処理をRubyで実装できることである。RubyにはRuby on Railsと呼ばれるプログラムの再利用の仕組みが存在する。既存のプログラムを再利用することによって，システムを最小限の労力で実装することができる。3つ目は，フレームワーク上で仮想スイッチを扱うことができるからである。

OpenFlowに対応した物理スイッチというものも販売されている。しかし，費用が数十万円程であるため低費用で実現するという目的と合致しない。そのため，ネットワーク上のスイッチをOpen vSwitchで仮想化し構築する。スイッチも仮想化することで，Linuxが動作する端末のみで動作させることが可能となる。これは，ソフトウェアでシステムのネットワークを構築でき，持ち運びなどの面においても有用である。

本システムのネットワーク構成図を図4に示す。また，本システムの全体像を図5に示す。本システムは，図5中央にあるLinux端末に実装した。

6. 実　　験

本システムを評価するために2つの実験を行なった。また，本システムに対する費用面での考察も合わせて行った。

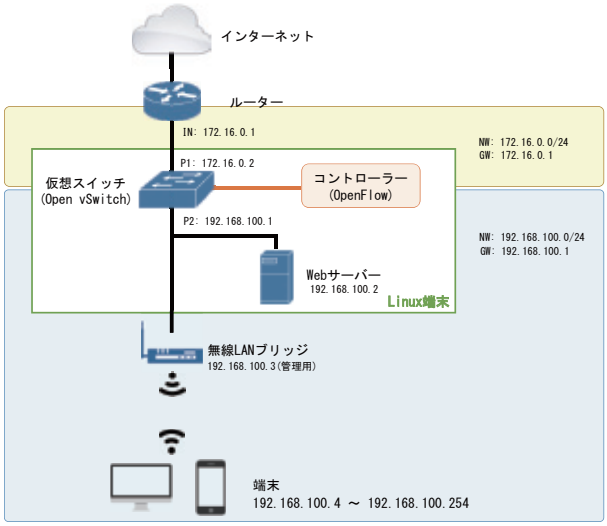

図4　ネットワーク構成図

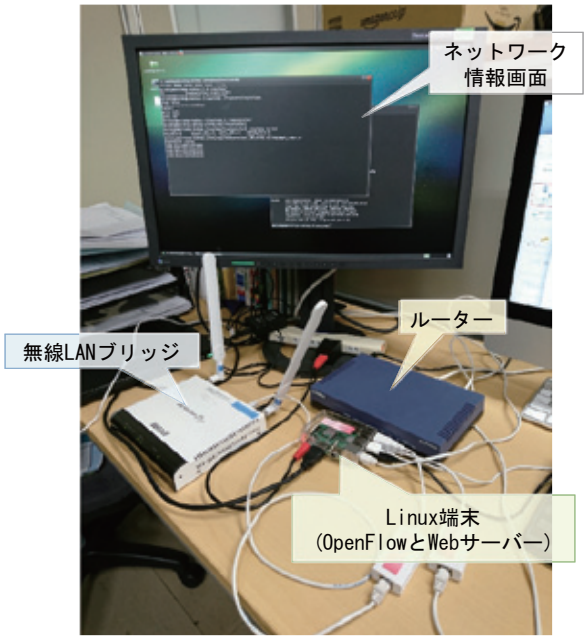

図5　システムの全景

6.1 【実験1】個人認証の有効性に関する実験

本システムで構築した仮想ネットワークに端末を接続し，通信の特定が可能であるかどうか，評価実験を行った。実験は，個人を識別可能かどうか，接続を拒否可能かどうか，これら2点を確認する。また，ネットワーク構築の経験が浅い人が使用できるかを確認するため，管理者側の操作についてアンケートを取得することとした。本システムは，国士舘大学の研究室内のLANに接続した。

以下に被験者5人による主観評価の結果を示す。実験には，各被験者が2台ずつ端末を接続すると仮定し，10台の端末を用意した。被験者はいずれも本学の理工学部理工学科電子情報学系の4年生である。ネットワークの

表3　個人認証の有効性検証実験結果

	出来た[人]	出来なかった[人]
(1)通信者の特定	5	0
(2)特定の通信の拒否	4	1

表4　本システムの使用評価

五段階評価(平均値)	3.2
コメント	こんな小さいシステムなら気軽に使えそう
	専門用語がよく分からない
	入力するだけなら簡単だ

構築経験はいずれの被験者も無かった。アンケートは，①「誰の通信か判断できたか」，②「特定の通信を拒否することができたか」についてである。さらに，システムの使いやすさに対して，5段階の評価を行った。また，選択肢の他に本システムに対しての自由記入欄を用意した。

表3，**4** より，個人を識別可能かどうか，接続を拒否可能かどうか，の2点においては概ね可能であると判断できる。特定の端末が拒否できないという理由に関しては，今回は通信の確認のためのパケット（ICMP）を拒否しない実装を行ったからであると推測できる。アンケートには，肯定的意見として「こんな小さいシステムなら気軽に使えそう」，「入力するだけなら簡単だ」，否定的意見として「専門用語がよく分からない」などの意見が寄せられた。これらの結果より，個人を識別するという面は達成されたが，簡便なシステムの構築という面は達成できなかったことがわかる。

既存のネットワークに接続する以上，ネットワークに関する最低限の専門知識が必要となる。本システムでは，テキスト形式の設定ファイルを編集する仕組みである。その際，ポートごとのIPアドレスやMACアドレス，ゲートウェイなどの設定が必要である。そのため，設置者に使いづらい印象を与えたのだと考える。これらは，端末のネットワーク環境を自動的に設定するDHCPや、IPアドレスとポートを変換する技術であるNAPTなどを導入することで緩和することが可能であると考える。

6.2 【実験2】ネットワークのレイテンシ測定実験

次に本システムを使用した際のレイテンシについて計測した。本システムは仮想化を行っているため，ネットワークの処理速度は端末の性能に影響を受ける。そのため，想定環境下で本システムを使用した際のレイテンシについて計測する必要がある。

本システムは会議での使用を想定しているため，10台の様々な端末が接続されている状況下で測定を行う。測定対象以外の端末は，任意のネットブラウジングを行っているものとする。また，その他の環境は実験1に準ずる。

表5　システム処理速度測定実験結果

	一般のルーター使用時の時間t[ms]	OpenFlow使用時の時間t'[ms]	時間差 (t'-t)[ms]	倍率 (t'/t)
1回目	30.2	20.7	-9.5	0.7
2回目	37.3	53.1	15.8	1.4
3回目	27.6	50.4	22.8	1.8
4回目	45.2	62.9	17.7	1.4
5回目	43.4	53.5	10.1	1.2
6回目	36.5	53.6	17.1	1.5
平均値	36.7	49.0	12.3	1.3

表6　使用機材一覧

使用機材	金額[円]
Raspberry Pi 3 Model B	5100
USB2.0 LANアダプター (LUA3-U2-ATX)	2180
micro SDHC 32GB (TS32GUSDU1PE(FFP))	980
2.4A USB急速充電器 (BSMPA2402P1BK)	954
microUSB充電専用ケーブル (OWL-CBJ5(B)-SP/U2A)	545
合計金額	**9759**

実験は端末から“google.co.jp”に対して通信を行い，どのぐらい時間がかかるのかを測定するものである。具体的には，端末からpingと呼ばれる回線状況を測定するコマンドを実行し，“google.co.jp”からレスポンスが来るまでの通信時間を測定する。送信するものは56［byte］のダミーデータである。なお，端末は本システムにより既に認証されているものとする。

表5 より，本システムを使用すると平均で12.3［ms］，最大で22.8［ms］遅延することが分かった。ネットワーク速度は，様々な要因により影響される。またWebページの閲覧に関しては，通信するデータ量が少ないため遅延の影響は受けにくい。したがって，20［ms］程度の遅延は特に問題はないと判断する。

本システムで使用したOpenFlowは，スイッチ上でパケットの可否を判断することができる。そのため，通常時よりもネットワークのレイテンシは低下する。しかし，認証の結果を直接フローテーブルに書き込むため，認証サーバへの接続待ちなどの問題は発生しづらいものと考える。以上より，レイテンシの面において，会議での使用は可能であると判断する。

6.3 本システムの費用について

最後に，本システムの費用面についての考察を行う。実装には安価なIoT機器であるRaspberry Pi 3を用いたため，1万円以下の費用での実装が可能であった。実際に発生した費用について，**表6** に示す。

これは市販されている認証システム搭載のネットワーク機器と比較して安価なものである。参考までにIEEE

802.1Xの認証システムを搭載したCONTEC社製のFX-SVR-RDSは現時点で約17万円である。

実際に企業がOpenFlowスイッチを試作した報告も存在する[11]。これは既存のスイッチ上にOpenFlowを実装したものである。OpenFlowは従来のネットワークレイヤーを超えた仕組みであるため実装の費用が増大する。対して本システムは，使用者が接続することのみに焦点を当てたため，費用の削減に繋がったと考える。

6.4 実験考察

本システムは，会議において一時的なネットワークを設営するものであった。既存のシステムの問題点として，設備費用と不正利用対策の2点を挙げた。6.1節より，本システムを使用して不正利用者の特定，除外が可能であることが判明した。また6.3節において，システム全体の費用として1万円以下の費用で構築できることを示した。また，6.2節の会議を想定した負荷実験において，処理能力も十分であることが分かった。以上より，本システムは，会議などで使用可能な臨時のネットワーク認証システムとなり得ると考えられる。

7. おわりに

本稿で提案したシステムは，臨時に行われる小規模の会議などにおいて，計算機ネットワーク環境を構築するものである。その際，不正利用の監視を行う必要があることを示し，その対策機能を装備した既存のネットワーク機器が高価であるため臨時に行われる小規模の会議には導入しづらいことを示した。これらの問題を，OpenFlowを用いてネットワークを仮想化し，ソフトウェアを用いて認証システムを実装することで解決した。6章に示した実験より，本システムを使用して不正利用者の排除が可能であることが分かった。通信遅延は発生するものの，これは20人規模では事実上問題とならないことを示した。また，本認証システムの費用は1万円程度に収まり，安価に実現できるシステムであることも示した。

一方，本システムの問題点としては，ネットワークに関する最低限の専門知識が必要となる点である。これについては今後の課題とするが，DHCPやNAPTなどを導入することで，より簡便なシステムを作ることができると考える。

参考文献

1) 日本経済新聞：無線LANのメール丸見え　成田・関西・神戸の3空港，入手先〈http://www.nikkei.com/article/DGXLASDG2600E_W4A820C1CR0000/〉(参照2017-7-19).
2) 後藤英昭，新妻共，大和純一：大規模学術系無線LANローミングのための集中型認証システム，電子情報通信学会誌，Vol.J100-D，No.5，pp.584-594（2017）.
3) 櫻田武嗣，三島和宏，萩原洋一：仮想化技術を活用した無線LANシステムの刷新，学術情報処理研究，No.18，pp.71-80（2014）.
4) 中村嘉志，瀬川典久，丸山一貴：ソフトウェアルーターを用いた学術会議のための一時的なインターネット接続基盤構築の実践～WISSの例～，情報処理学会 デジタルプラクティス，Vol.7，No.4，pp.407-416（2016）.
5) 松谷健史：ARPを利用したローカルエリアネットワークにおける不正接続の排除，マルチメディア通信と分散処理ワークショップ論文集，Vol.2004，No.15，pp.221-225（2004）.
6) 会議室.com：会議室.com，入手先〈https://www. kaigishitu.com〉(参照2017-7-15).
7) Nick, M., Tom, A., Hari, B., et al.,: OpenFlow: Enabling Innovation in Campus Network, ACM SIGCOMM Computer Communication Review, Vol.38, pp.69-74（2008）.
8) 高宮安仁，鈴木一哉，松井暢之，ほか：OpenFlow実践入門，ISBN：978-4-7741-7983-4，技術評論社（2013）.
9) Cucumber Limited：Relish Project:Trema，入手先〈https://relishapp.com/trema/trema/docs〉(参照2016-11-16).
10) Linux Foundation Collaborative Projects：Open vSwitch，入手先〈http://openvswitch.org〉(参照2016-11-16).
11) 千葉 靖伸：OpenFlowスイッチの試作と評価，新世代ネットワークワークショップ2009予稿集，pp.145-150（2009）.