OpenFlowとShibboleth認証を用いた利用者認証システムの開発

山下 翔平[1,a)]　田中 久治[1]　堀 良彰[2]　大谷 誠[3]　渡辺 健次[4]

概要：大学のように開かれた場所では，ネットワークの利用時に認証を行い，構成員のみに利用させる必要がある．佐賀大学ではネットワークの認証にShibbolethによるシングルサインオン認証を導入している．一方，近年，SDN（Software Defined Network）という概念が注目されている．これはソフトウェアによってネットワーク全体を制御しようという考え方である．そのSDNの標準として注目されている技術としてOpenFlowがある．OpenFlowを用いることで，プログラミングによってパケットの制御が可能になる．我々はOpenFlowによるパケット制御とShibboleth認証を用いたネットワーク利用者認証システムを開発した．

キーワード：OpenFlow, Shibboleth

Development of Network User Authentication System using OpenFlow and Shibboleth Authentication

Abstract: Although a university is a public space, only members can use the network of the university. Therefore, when members use the network, it is necessary to authenticate. We are using Single Sign-On authentication by shibboleth authentication in Saga University. On the other hand, the concept of SDN (Software Defined Network) came out. This is a way of thinking that we can control a network by software. "OpenFlow" is new implementation to control a network and well known as a standard of SDN recently. In this research, we have developed a flexible network user authentication system using shibboleth authentication and OpenFlow.

Keywords: OpenFlow, Shibboleth

1. はじめに

大学のように開かれた場所では，ネットワークの利用時に認証を行い，構成員のみに利用させる必要がある．佐賀大学ではネットワークの利用者認証システムとしてOpengateを使用しており，Shibbolethによるシングルサインオン認証を導入している．

一方，近年，SDN（Software Defined Network）という概念が注目されている．これはソフトウェアによってネットワーク全体を制御しようという考え方である．そのSDNの標準として注目されている技術としてOpenFlowがある．OpenFlowを用いることで，プログラミングによってパケットの制御が可能になる．

佐賀大学で使用しているOpengateはL3で動作し，導入にはネットワークを区切る必要がある．本研究では，OpenFlowによるパケット制御とShibbolethによるシングルサインオン認証によってネットワークを区切ることなく，L2スイッチのように導入が可能なネットワーク利用者認証システムを開発した．

[1] 佐賀大学大学院工学系研究科
Graduate School of Science and Engineering, Saga University
[2] 佐賀大学全学教育機構
Organization for General Education, Saga University
[3] 佐賀大学総合情報基盤センター
Computer and Network Center, Saga University
[4] 広島大学大学院教育学研究科
Graduate School of Education, Hiroshima University
[a)] yamashita@ai.is.saga-u.ac.jp

2. 基礎技術

2.1 OpenFlow

OpenFlow とは，スタンフォード大学に在籍していた Martin Casado 氏による研究が発端となって開発されたネットワーク制御技術である．現在は，Open Networking Foundation（ONF）が中心となって標準化と普及が推進されている．この OpenFlow の基本的な考え方は，これまで 1 台のスイッチ (あるいはルータ) に統合されていたフレームの転送機能と経路制御機能を分離し，オープンなインタフェースでつなぐものである．経路制御機能を OpenFlow コントローラと呼び，転送機能を OpenFlow スイッチと呼ぶ．本研究では，OpenFlow プロトコルのバージョンは 1.0.0 を採用している．

2.1.1 OpenFlow コントローラ

OpenFlow コントローラとは，OpenFlow において，経路制御機能をもつソフトウェアである．処理方法と経路を計算し，OpenFlow スイッチへと送信する．送信の際には，何にその処理方法を適用するか，という条件を追加する．この条件と処理方法の組をフローエントリと呼び，フローエントリは OpenFlow スイッチ中のテーブルに格納される．OpenFlow コントローラの開発フレームワークには Trema を使用している [1], [2].

2.1.2 OpenFlow スイッチ

OpenFlow スイッチとは，OpenFlow において，パケットの転送機能をもつソフトウェアである．OpenFlow スイッチはフローエントリを確認し，条件に合うパケットに対して，指定された動作を行う．OpenFlow スイッチのソフトウェアとして OpenFlow 1.0 for OpenWRT を使用している [3], [4].

2.2 Shibboleth

Shibboleth はプロジェクトの名称であり，次世代インターネット研究開発コンソーシアム「Internet2」において研究が進められている．Shibboleth 認証は，セキュアなシングルサインオンの仕組みであり，複数サービスへのログインを共通化してスムーズに利用することができる．一度ログインしておくと認証が継続するので，対応している他のサービスを使った場合，ID とパスワードを再度入力する必要がない．つまり，Shibboleth 認証で 1 度ログインすることで学内ネットワークはもちろん，学内に存在しているアプリケーションにも再度ログインすることなく利用することができる．Shibboleth 認証は IdP（Identity Provider）と SP（Service Provider）から構成される．

2.2.1 IdP（Identity Provider）

IdP はディレクトリサービスやデータベースを用いてユーザの情報を保持している．SP から認証リクエストを受け取るとユーザ認証を行う．認証に成功したユーザに対してアサーションを発行し，SP へと処理を返す．

2.2.2 SP（Service Provider）

SP はリソースを保護し，リソースへとアクセスするユーザに認証を行わせるために IdP へと転送する．アサーションを受け取ると，ユーザの属性情報をもとにリソースへアクセスする資格があるかチェックする．

2.3 Opengate

Opengate は佐賀大学で開発され，運用されているネットワーク利用者認証システムである．不特定多数のネットワーク利用者が様々な端末を接続するネットワーク環境において，利用者認証と利用記録を行うことができる．このシステムは，端末群と利用ネットワークとの間にゲートウェイを設置し，そこを通過するパケットをフィルタリングする．Opengate では，ネットワークの利用資格と Web ブラウザさえあれば，特別な申請やソフトウェアの準備をすることなく，端末をインターネットに接続できる [5].

Opengate の構成を**図 1** に示す

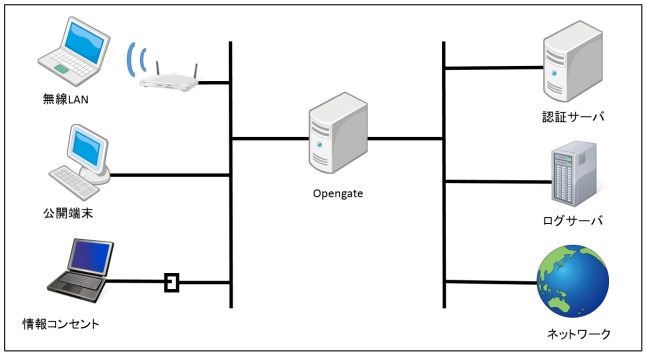

図 1 Opengate の構成
Fig. 1 Configuration of Opengate

Opengate の動作の流れを以下に示す．

(1) 利用者が公開端末や，持参 PC で任意の Web サイトへアクセスしようとする．
(2) Opengate はこの通信を横取りし，ユーザ ID とパスワードを要求するネットワーク利用者認証ページを返す．
(3) 利用者が入力した情報を認証サーバに送る．
(4) 認証に成功すると，当該端末に対してファイアウォールを開放しネットワークに利用が可能になる．
(5) 利用者監視ページを端末に送り，利用終了を監視する．
(6) 定期的な生存確認を行い，生存が確認できなければファイアウォールを閉鎖し，記録をとる．

3. システムの概要

本システムは OpenFlow と Shibboleth 認証を用いてネットワーク利用者の認証と通信路の制御を行うシステムであ

る．システムの動作は Opengate をもとにしているが，L2 で動作する．OpenFlow コントローラ上でパケット制御モジュールが動作し，IP アドレスや MAC アドレスといった端末の情報と利用許可の有無をデータベースに保持している．なお，制御モジュールの開発言語には Ruby[6]，データベースには SQLite3 を用いている [7]．またデータベースとは別にユーザの利用状況を把握するためのログ機能がある．開発環境を**表 1** に示す．

表 1 開発環境
Table 1 Development Environment.

分類	詳細
コントローラ OS	Ubuntu 12.04
コントローラ開発フレームワーク	Trema 0.2.5
OpenFlow スイッチ	OpenFlow 1.0 for OpenWRT
データベース	SQLite3 3.7.9
開発言語	Ruby 1.8.7, JavaScript
Shibboleth IdP サーバ	2.1.5
Shibboleth SP サーバ	2.4.3

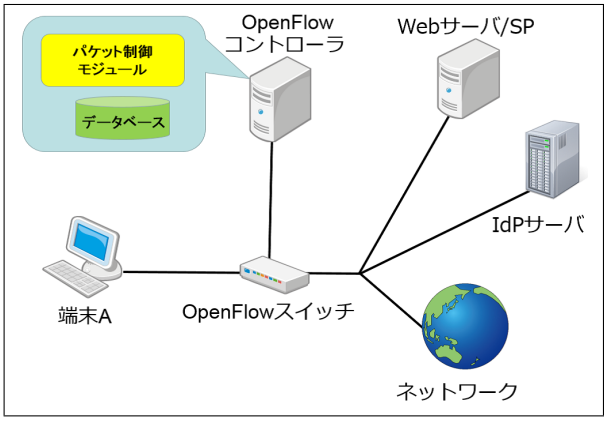

図 2 システム構成
Fig. 2 Operation flow

システムの構成を**図 2** に示す．OpenFlow コントローラ上ではパケット制御モジュールとデータベースが動作している．図 2 を参考に，システムの流れを以下に示す．

(1) ネットワークを利用したい端末 A から OpenFlow スイッチに対して HTTP パケットが届く．
(2) OpenFlow スイッチは端末 A に対するフローエントリがあるかフローテーブルを確認する．
(3) フローエントリがある場合，OpenFlow スイッチは OpenFlow コントローラに対して処理方法を問い合わせる．
(4) OpenFlow コントローラはデータベースを調べ，端末 A がネットワークの利用許可を持っているか確認する．
(5) 利用許可を持っていない場合，OpenFlow コントローラは OpenFlow スイッチに対して，Web サーバ/SP に宛先を書き換えるように命令する．
(6) Web サーバ/SP はパケットを受け取ると，IdP サーバへと認証リクエストを送信する．
(7) IdP サーバから端末 A に対して認証ページが返される．ユーザはユーザ ID とパスワードを入力し認証を行う．
(8) 認証に成功すると Web サーバ/SP は端末 A に認証成功ページを返す．この際，Web サーバ/SP は端末 A に対して，宛先ポート番号が 10000 番のパケットを送信する．
(9) OpenFlow スイッチは OpenFlow コントローラに端末 A が認証に成功したことを伝え，(8) で作成されたパケットを削除する．
(10) OpenFlow コントローラは端末 A の利用許可を変更し，ネットワークを利用できるようにする．
(11) OpenFlow コントローラは利用許可を持つ端末に対して，MAC アドレスを条件としたフローエントリを作成する．
(12) フローエントリ作成後は自由にネットワークを利用できる．
(13) ブラウザが閉じられると Web サーバ/SP は端末 A に対して，宛先ポート番号が 20000 番のパケットを送信する．
(14) OpenFlow スイッチは OpenFlow コントローラに端末 A がネットワークの利用を終了したことを伝え，(13) で作成されたパケットを削除する．
(15) OpenFlow コントローラは端末 A の利用許可を変更し，ネットワークの利用許可を奪う．
(16) OpenFlow コントローラは利用許可を失った端末に対して，フローエントリを削除する．
(17) OpenFlow コントローラは一定時間毎に ARP リクエストを送信し，返信がない端末については閉鎖処理を行う．

4. システムの利用

本システムを利用するにあたり，パケット制御モジュールと OpenFlow スイッチを起動する必要がある．パケット制御モジュールを起動し，OpenFlow スイッチを起動すると，接続したスイッチの ID が表示される．この状態で OpenFlow スイッチに接続した端末でブラウザを開くと認証ページが表示される (**図 3**)．

ID とパスワードを入力し認証に成功すると，認証成功画面（**図 4**）が表示され，ネットワークを利用できるようになる．認証成功画面を閉じることで利用が終了となり，利用許可の削除が行われる．認証成功画面を開いてから閉じるまでが利用時間としてログに記載される．ログには利用者 ID，時刻，ブラウザ情報等が記載される．

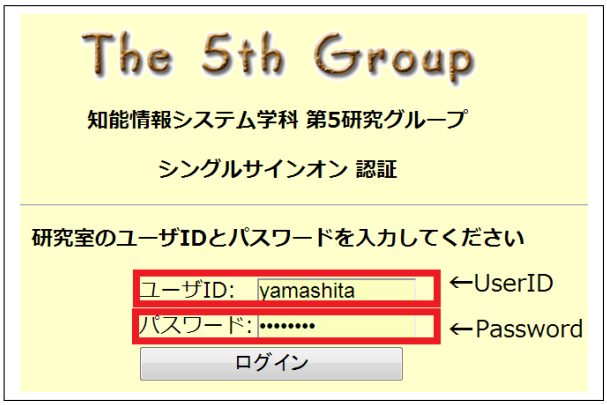

図 3 認証ページ
Fig. 3 Authentication page

図 4 認証成功画面
Fig. 4 Authentication success page

5. 動作検証

以下の OS とブラウザでシステムの閉鎖処理が動作することを確認した.

- OS
  Windows7（64 ビット OS）

- ブラウザ
  Internet Explorer 10.0
  Firefox 25.0
  Google Chrome 31.0

6. まとめと今後の課題

本研究では OpenFlow と Shibboleth を用いて，新たな利用者認証システムを開発した．L2 スイッチのように、ネットワークを区切ることなく導入することができる．また，Shibboleth 認証を用いることにより，シングルサインオンにも対応した.

今後の課題は，IPv6 への対応である．強制遷移の処理部分でのみ IP アドレスの処理が必要である．現在利用している，OpenFlow v1.0 では IPv6 を扱うことが出来ず，対象が IPv4 ネットワークに限られてしまう．今後は IPv6 ネットワークが急速に普及していくことが考えられるので，OpenFlow v1.2 以降を採用し，IPv6 に対応させる必要がある.

参考文献

[1] Trema Full-Stack OpenFlow Framework in Ruby and C(online)，入手先 ⟨http://trema.github.io/trema/⟩
[2] 高宮安仁, 鈴木一哉：新ネットワーク制御技術 OpenFlow 実践入門，技術評論社 (2013).
[3] Pantou : OpenFlow 1.0 for OpenWRT(online)，入手先 ⟨http://archive.openflow.org/wk/index.php/Pantou_:_OpenFlow_1.0_for_OpenWRT⟩
[4] OpenFlow in theBox(online)，入手先 ⟨http://openflow.inthebox.info/Documents⟩
[5] シングルサインオンに対応したネットワーク利用者認証システムの開発: 大谷誠, 江藤博文, 渡辺健次, 只木進一, 渡辺義明, 情報処理学会論文誌, Vol.51, No.3, pp.1031-1039 (2010)
[6] RubyDoc.info(online), http://rubydoc.info/github/trema/trema/master/frames
[7] SQlite Home Page(online), 入手先 ⟨http://www.sqlite.org/⟩