ユーザ利便性向上を目指した IoT のための SDN セキュリティポリシー設定機構

SDN Security Policy Setting mechanism of IoT for User Conveniences

吉村 悠 † 佐藤 健哉 ‡
Haruka Yoshimura Kenya Sato

1. はじめに

近年，情報家電など IoT 機器の普及が進みインターネットに繋がる家庭内のデバイスや提供するサービスが増加し，家庭内のネットワークが複雑化している．こういったホームネットワークでは，セキュリティが不十分なまま IoT 機器が接続されている場合があり，2016 年にはマルウェアに感染した家庭内の IoT 機器を踏み台とした，大規模 DDoS 攻撃が発生するなど [1] ホームネットワークのセキュリティ問題への注目が高まっている．ホームネットワーク特有の問題として、ユーザのセキュリティ知識や意識が低いために [2]，適切なネットワーク設定ができなかったり，IoT 機器への攻撃に気づかないといった事が挙げられる．また何らかのセキュリティシステムを用意してもユーザが使用しないといった問題もある．対策としてネットワーク機器がユーザの代わりに自動でネットワーク設定を行うことが考えられるが，ユーザの意図しない設定が行われたり，ユーザの意思を設定に反映するのが難しい．

本研究では，Software-Defined-Networking（SDN）の代表的なプロトコルである OpenFlow を用いて，自動でネットワーク設定を作成し，それをユーザが理解しやすいセキュリティポリシーに変換してユーザに提案し，ユーザの承認を得てネットワーク設定を行うことで，ユーザの負担を軽減しつつ動的にユーザがセキュリティルールを設定できる機構を提案する．

2. 関連研究

Lee ら [3] は，IT 知識のないホームネットワークユーザ向けに SDN を利用したネットワーク自動設定システムを提案している．ホームネットワークに接続されたデバイスの IP と MAC アドレスからデバイスデータベースを構築し，接続されたデバイスの通信制御を行うことでデバイスごとの通信品質の制御など行う．しかしセキュリティについての具体的な対策はしていない．村上ら [4] は，ホームネットワークの不正アクセス対策として，OpenFlow による認証技術を提案している，この中でホームネットワーク内には異なる規格のデバイスやそれに搭載される様々なアプリケーションが混在しているため，セキュリティシステムはすべての端末や規格に対応したソフトウェアとして構築するのではなく，ホームネットワーク内のデバイスが必ず利用するネットワークのシステムとして構築することが望ましいと述べ，OpenFlow を利用している．柴田ら [5] は，ホームネットワーク内のデバイス管理のために OpenFlow によってネットワークを仮想的に分離してスライスとして管理する方法を提案している．所属するスライスが異なるデバイス間の通信は全て遮断されるので，デバイス間の不正通信を抑制できる．しかし各スライスはデバイスのベンダー番号により構成されるので，スライスは静的で，他社製品間の通信はできないなどユーザの意思の反映ができない．

本研究では，仮想的に分離されたデバイスが他のデバイスと通信する際に，通信を許可するネットワーク設定を自動で作成し，その承認をユーザに求めてから実際に設定する．その際 UPnP（Universal Plag and Play）などのデバイス間調停プロトコルを利用して得たデバイスや提供するサービスの情報を用いて，ユーザにどういった通信設定の承認を求めているか，より具体的に伝えることでユーザの負担を軽減しつつ，ユーザが適切にネットワーク設定を操作できるようにする機構を提案する．

3. 提案機構

3.1 概要

提案機構の概要を図 1 に示す．提案機構では，各デバイスはネットワーク上で仮想的に分離されており，デバイス間の通信は遮断されている．デバイスの通信開始時にそのコネクションの通信を許可するネットワーク設定を自動で作成し，ユーザに設定の承認を求める．その際事前に収集した，該当のデバイス，サービス情報を用いて，どういったネットワーク設定の承認を求めているかを具体的に表すセキュリティポリシーとしてユーザに提案する．ユーザはそのセキュリティポリシーが使用状況に即したものか判断し，可否の返答をする．返答を受け提案機構は実際にネットワークの設定または棄却を行う．

3.2 ネットワークの仮想的な分離

一般にホームネットワークは単一のネットワークに異なる用途のデバイスが接続した形態となっているが，このようなネットワーク構成でネットワーク設定を行う場

‡ 同志社大学大学院理工学研究科情報工学専攻，
Guraduate School of Information and Computer Science,
Doshisha University

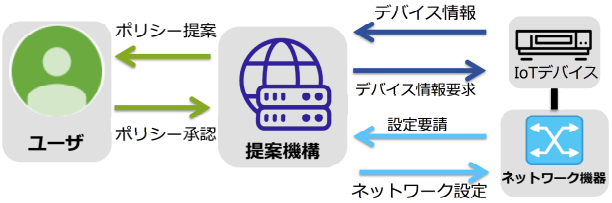

図 1　提案機構概要

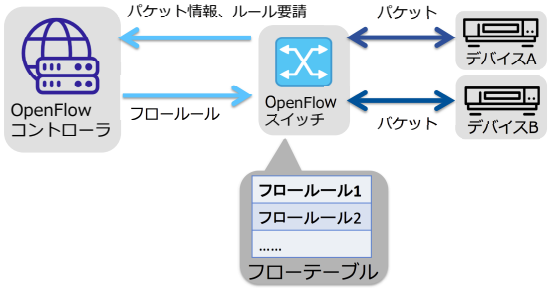

図 2　OpenFlow による通信制御例

合，ある特定のデバイスのために行ったネットワーク設定が他のデバイスに影響を与える可能性があり，セキュリティリスクになりうる．対策として VLAN などで一つのネットワークを仮想的に分離し，分離したネットワークごとに設定を行う方法がある．ただし分離した各ネットワークに各デバイスを用途に合わせて適切に接続する必要があるなど，ネットワーク設定は複雑となり管理コストは大きくなる．

3.3　通信制御

本研究では，提案機構のネットワーク通信制御に OpenFlow を利用する．OpenFlow とは SDN の代表的プロトコルの一つであり，ネットワークをソフトウェアで制御する技術である.OpenFlow ネットワークの例を図 2 に示す．従来のネットワーク機器のデータ伝送部と経路制御部をそれぞれ OpenFlow スイッチ（以下スイッチ）と OpenFlow コントローラ（以下コントローラ）に分離した構成となっている. コントローラが各スイッチのフローテーブルに，どういったパケットをどう制御するかを示すフロールールを設定することで通信を制御する．フロールールは通信パケットのレイヤー 2 からレイヤー 4 までの情報を扱うことができ，ネットワークの仮想的な分離などを柔軟に行える．またフロールールをコントローラが自動で作成，設定することでネットワーク構成の動的な制御が可能となる．提案機構では，OpenFlow ネットワークによって各デバイスを仮想的に分離すると共に，デバイスが通信する際にデバイス間のネットワークを繋げるフロールールをコントローラが自動作成し，ユーザがその設定の認可を行うだけでネットワーク構成を制御できるようにする．

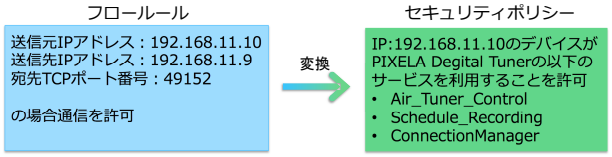

図 3　セキュリティポリシー変換例

3.4　デバイス情報収集

本研究では，UPnP を使用してホームネットワークのデバイス除法を収集する．UPnP は業界団体 UPnP Forum が標準仕様を策定する IoT デバイス間調停用プロトコルであり [7] 主に AV 機器などに使用されている．HTTP などを用いて，あるデバイスが他のデバイスに対し自身のデバイス情報や提供するサービス情報を通知し，デバイスがネットワークに接続されるだけで他のデバイスから利用可能な状態にすることができる．提案機構では，この UPnP のデバイス情報の通知を収集しておき，ユーザにフロールールの認可を求める際にそのフロールールにデバイス情報を加え具体化したセキュリティポリシーとして提案する．

3.5　セキュリティポリシー

広義では情報セキュリティ対策について、総合的・体系的かつ具体的にとりまとめたものを表す．本研究では，どのデバイスを誰にアクセスさせ、誰にアクセスさせないか示すものとする．OpenFlow 機構が自動で作成した，フロールールをこのセキュリティポリシーに変換して具体的にユーザに提案することでユーザが適切に判断できるようにする．変換の具体例を図 3 に示す．なお以下ではセキュリティポリシーをポリシーと呼ぶ．

3.6　構成

提案機構は主に以下の 4 つの動作から成る．各動作の概要を以下に示す．

デバイス情報収集

ホームネットワーク内のデバイス情報と提供するサービス情報を収集，保存する．本研究では UPnP を利用する．

ポリシー作成

デバイスが通信する際にフロールールを作成し，収集したデバイス情報とサービス情報を合わせ，セキュリティポリシーを作成する．

ポリシー提案

作成されたポリシーをユーザに提案し，可否の返答を受け取る．

ポリシー設定

ユーザの返答から該当するフロールールの設定また

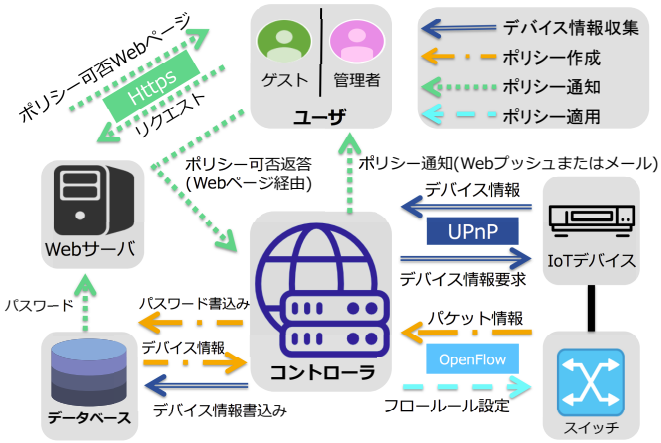

図 4　提案機構のモジュール構成

図 5　デバイス情報収集モジュール

は棄却を行う．

3.7　モジュール構成

上記の 4 つの動作をそれぞれモジュールとして実装する．提案機構のモジュール構成の概要を図 4 に示す．各モジュールの詳細を以下に示す．

3.8　デバイス情報収集モジュール

UPnP 対応機器は M-search メッセージという UPnP で規定されたメッセージを受け取ると自身のデバイス，サービスの詳細を得るための URL を返信する．この M-search メッセージを一定時間ごとにホームネットワーク内にマルチキャストし（図 5-1）返信された URL から詳細情報を収集し（図 5-2）データベースに保存する（図 5-3）．詳細情報は，IP 番号や TCP 番号に加え，UPnP Forum によって定義されたデバイスタイプ，サービスタイプ，製造者名，サービスを利用するための URL などである．

3.9　ポリシー作成モジュール

ポリシー作成モジュールはコントローラ内に実装される．まだフロールールで通信設定がされていないデバイス間が通信を始めると，スイッチがそのパケットをコントローラに転送し（図 6-1）フロールールを要請する．このモジュールはパケットの情報からデータベースを検索し，通信を行うデバイスの情報を取り出し（図 6-2），セキュリティポリシーを作成する（図 6-3）．また ID とパス

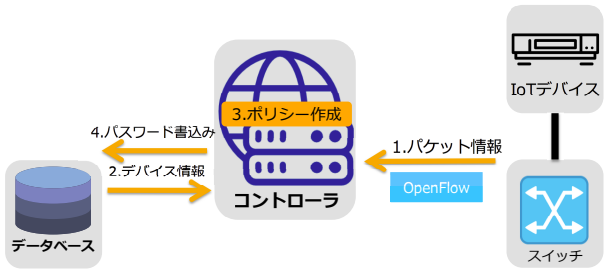

図 6　ポリシー作成モジュール

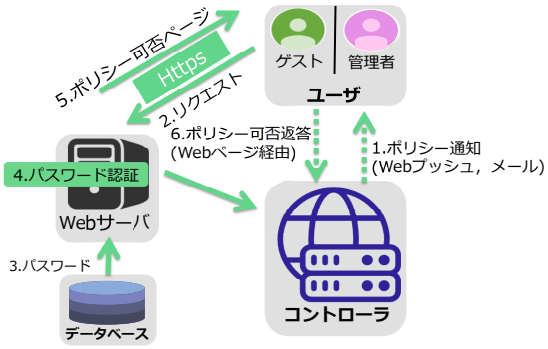

図 7　ポリシー提案モジュール

ワードを作成しセキュリティポリシーと対応させてデーテベースに保存しておく（図 6-4）．これはユーザがポリシー可否応答を行う際のアクセス制御に使用する．

3.10　ポリシー提案モジュール

作成されたポリシーをユーザに提案し返答を受け取る．ユーザへの提案はメール，または Web プッシュ機能により行い（図 7-1），提案メッセージには，ポリシー可否 Web ページへのリンクが含まれユーザがアクセスする（図 7-2）．アクセスすると URL パラメータから ID とパスワードを読み取り，ポリシー作成時に保存したパスワードと照合する（図 7-3，4）．照合されるとポリシー可否 Web ページが表示され（図 7-5），ページ上の可否ボタンをクリックするとポリシー設定モジュールにその結果が送信される（図 7-6）．ユーザには管理者ユーザとゲストユーザの分類があり，ポリシーがインターネットへの通信を許可するものかどうかによって，管理者ユーザのみに提案するか両方に提案するか選択され，提案方法も異なる．提案方法は以下の 2 つである．

管理者ユーザへのメールによる提案

管理者ユーザとして事前に登録されたメールアドレスに提案メールを送る．インターネットへの通信設定の場合はこの提案のみ行われる．メッセージの URL から，ホームネットワーク外部からもポリシー可否を行える．

ゲストユーザへの Web プッシュ提案

ホームネットワーク内の登録ブラウザに対し Web

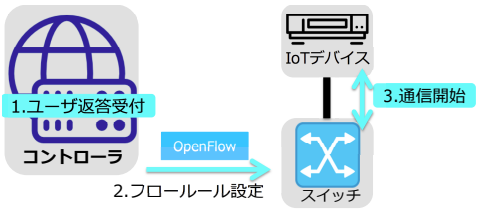

図 8　ポリシー設定モジュール

図 9　提案メッセージ

プッシュによる提案を行う，ブラウザ登録は，ホームネットワーク内からなら登録 Web ページで誰でも行える．このメッセージの URL はホームネットワーク内からのみ使用できる．今回は Google のプッシュサーバを利用するため，使用ブラウザは Chrome に限定される．

管理者ユーザに送られる提案メッセージを図 9 に，ユーザがアクセスするポリシー可否ページを図 10 に示す．なおゲストユーザへの Web プッシュ提案メッセージは図 9 から LAN 外からのアクセス URL を省いたものとなっている．

3.11　ポリシー設定モジュール

ユーザからの返答を受けて（図 8-1），承認された場合はスイッチに該当のフロールールの設定を行う（図 8-2），棄却された場合，該当するフロールールをリセットする．これらは既に他のユーザが返答を行っていた場合，上書きされるが管理者ユーザが設定したフロールールはゲストユーザには上書きされない．フロールールが設定されると，デバイスは規定された通信が可能となる（図 8-3）．

4.　動作手順

提案機構のセキュリティポリシー作成、フロールール設定までの動作手順を以下に示す．またフローチャートを図 11 に示す．

図 10　ポリシー可否ページ

1. デバイス情報収集モジュールがホームネットワーク内のデバイスや提供するサービス情報を収集しする．
2. あるデバイスが通信を開始する
3. スイッチがコントローラにパケットを転送しフロールールを要請する
4. ポリシー作成モジュールがパケット情報と収集したデバイス情報からセキュリティポリシーを作成しポリシー提案モジュールへ渡す．
5. ポリシー提案モジュールがフロールールによって提案するユーザを選択する．インターネットとの通信の場合は管理者ユーザに，ホームネットワーク内での通信の場合は管理者ユーザとゲストユーザを選択する．
6. 管理者ユーザへの場合はメールで，ゲストユーザへの場合は Web プッシュによる提案を行う
7. 提案を受けたユーザは，提案メッセージ内の ID とパスワードパラメータ付きのリンクにアクセスする．
8. リンク先のポリシー可否ページでは ID とパスワードのパラメータから正規のアクセスか判断した後，返答ボタンを表示しユーザの返答結果をポリシー設定モジュールに返す．
9. 返答を受けて，ポリシー設定モジュールがフロールールを設定または棄却する．
10. 既に他のユーザが返答をしていた場合，フロールールを上書きする．管理者ユーザのフロールールは優先され上書きされない．

5.　実装・評価

5.1　実装

実装環境を図 12，表 1 に示す．今回スイッチは Raspberry Pi3 に OpenVswitch をインストールし実装し，Rasbberry Pi3 の USB ポートに USB 変換 LAN アダプタを用

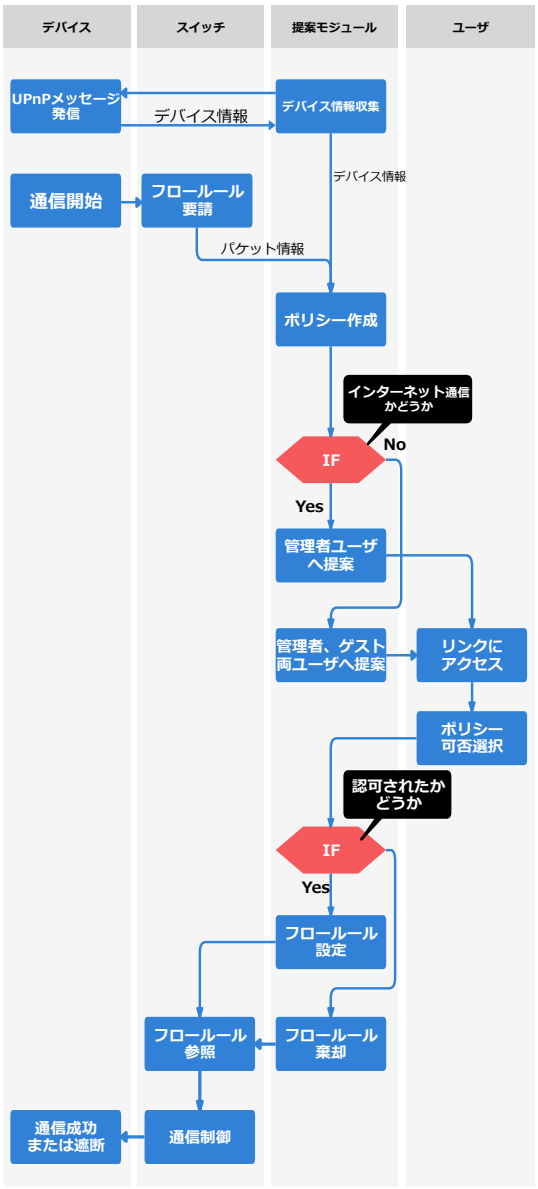

図 11　動作フローチャート図

いてスイッチの LAN ポートとしている．コントローラは UbuntuOS 上に Trema を使用して実装している．IoT デバイスとして録画ビデオの配信機能を持つデジタルチューナーを使用した．家庭用ルータとホームネットワークの間に提案機構スイッチを入れて提案機構をホームネットワークに導入する構成となっている．DHCP や NAT など，一般にホームネットワークに必要とされるネットワーク機能は家庭用ルータが提供する．また Web サーバへのアクセスなど，提案機構が必要とする通信を許可するフロールールを事前に設定しておく．

5.2　評価

性能評価として，ユーザがホームネットワーク内からポリシー可否ページ上で返答してから，フロールールが設定するまでの動作を 10 回行い，その処理時間を計測

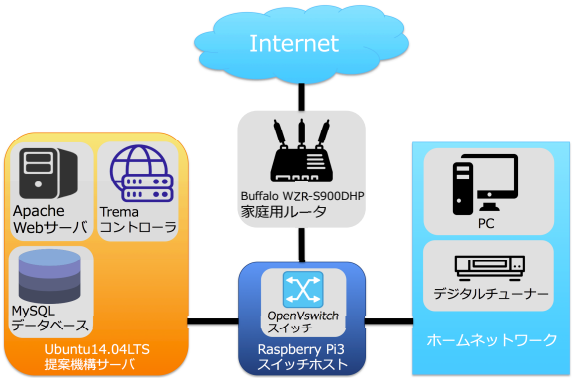

図 12　実装環境

表 1　実装環境

項目	バージョン
OpenFlow コントローラ	Trema Version 0.10.1
データベース	MySQL 5.7.19
Web サーバ	Apache 2.4.18
OpenFlow スイッチ	Open vSwitch Version 2.0.2
提案機構サーバ OS	Ubuntu14.04LTS
スイッチホスト OS	raspbian 8.0

した（表 2），またホームネットワーク内のデバイス間の通信のラウンドトリップタイムと，ホームネットワークデバイスからインターネット上のサーバ間の通信におけるラウンドトリップタイムとスループットを同様に計測し従来手法と比較した．(図 13)．従来手法と比べ，ラウンドトリップタイムが増加しスループットが低下した．ホームネットワークデバイス間の通信では約 3 倍，ホームネットワークデバイス・インターネット間では約 1.03 倍にラウンドトリップタイムが増加し，スループットは前者が約 5 分の一，後者が 4 分の 1 まで低下した．これはスイッチのソフトウェア化によるオーバヘッドに加えて，実装したラズベリーパイと家庭用ルータとのハードの性能差も関係している．

6.　考察

6.1　提案機構について

提案機構によってユーザの負担を減らしつつ，ユーザによるネットワーク設定が可能となる．これによってセキュリティの強化に加えて，ユーザの意思をネットワーク設定に反映でき，ユーザの意図しないネットワーク設定や通信を防ぐことができる．提案機構ではシステム面では検知できない攻撃であっても，その時に使用していないデバイスからの通信要請などはユーザの判断で遮断できる可能性がある．またポリシー提案型の機構により

表 2　フロールール設定までの動作時間（ミリ秒）

平均値	最小値	最大値
365.0	198.3	891.7

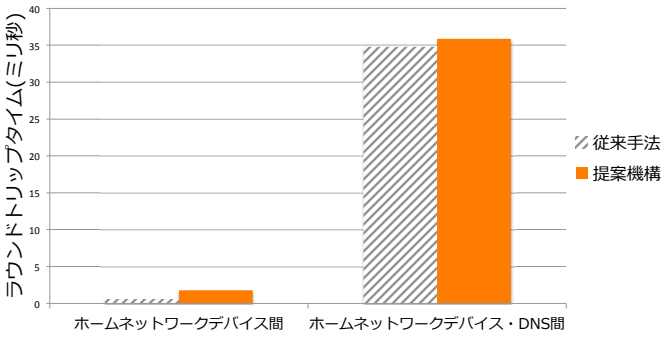

図 13　ラウンドトリップタイムの測定結果

セキュリティ意識の低いユーザであっても，デバイス使用時に必ずネットワーク設定を行うので設定の漏れを無くすことができる．

今回は UPnP によるデバイス情報のデバイスタイプ，サービスタイプを収集しセキュリティポリシーに記載したが，これに準ずる情報を収集できるなら，他のプロトコルを使用しても他のモジュールに影響は与えない．

6.2　評価結果について

従来手法とラウンドトリップタイムとスループットによる比較を行ったところ，提案機構は通信性能が大きく低下したが，これは実装したスイッチのハードウェアの性能差による部分もあるためさらなる検証が必要である．

6.3　今後の課題

提案機構を導入した場合，スイッチのオーバヘッドによる通信性能の低下が見られたので，全通信をスイッチに経由させるのではなく，必要な通信のみスイッチに経由させ，認可されたコネクションはそのままルータで通信するなど対策が必要である．

7.　まとめ

本研究では，複雑化するホームネットワークのセキュリティをテーマとして，特にユーザのセキュリティ知識や意識の低さから適切な設定がてきない事を問題点として挙げた．提案機構では OpenFlow を用いた SDN によりデバイスは仮想的に分離され，デバイス間の通信を許可するフロールールが設定されないとデバイス間の通信はできないようにして，通信時にフロールールを OpenFlow コントローラが自動で作成し，設定する際にユーザにどういったルールなのかを具体的に表した，セキュリティポリシーとして提案し，可否返答を受けてフロールール

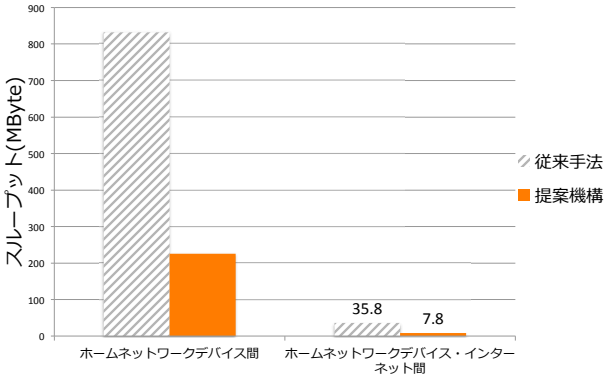

図 14　スループットの測定結果

の設定または棄却を行うことで，提案機構によってユーザの負担を減らしつつ，ユーザによるネットワーク設定を可能とする．提案機構によってセキュリティの強化に加えて，ユーザの負担を軽減しつつユーザの意思をネットワーク設定に反映することができた．

謝辞

本研究の一部は JSPS 科研費 16H02814 の助成を受けたものである．

参考文献

1) TRENDmicrio，"マルウェア「Mirai」による DDoS 攻撃が多発"，https://www.is702.jp/news/2050/，2017

2) 野村総合研究所，"情報家電ネットワークの現状と課題"，https://www.ipa.go.jp/files/000014114.pdf，2017

3) Lee Minseok，Younggi Kim，Younghee Lee. "A home cloud-based home network auto-configuration using SDN." Networking, Sensing and Control (ICNSC), 2015 IEEE 12th International Conference on. IEEE, 2015.

4) 村上萌, 中村嘉隆，高橋修. "OpenFlow を用いたホームネットワークへの接続端末制御による不正アクセス防御手法の提案." 研究報告マルチメディア通信と分散処理 (DPS) 2016.29 (2016): 1-6.

5) 柴田尚紀，山崎憲一. "B-16-9 SDN 技術を導入したホームネットワーク環境 (B-16. インターネットアーキテクチャ, 一般セッション)." 電子情報通信学会総合大会講演論文集 2014.2 (2014): 561.

6) McKeown, Nick, et al. "OpenFlow: enabling innovation in campus networks." ACM SIGCOMM Computer Communication Review 38.2 (2008): 69-74.

7) UPnP-forum，"UPnP-arch-DeviceArchitecture-v1.1"，http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.1.pdf，2017