情報処理学会論文誌　コンシューマ・デバイス & システム　Vol.8　No.2　34–42 (May 2018)

コンシューマ・デバイス論文

Bluetooth over DTLSによるIoTデバイスの遠隔制御システム

岡田 真実[1,†1,a)]　鈴木 秀和[1,b)]

受付日 2017年9月30日, 採録日 2018年2月14日

概要：IoT デバイスを制御するための短距離無線通信規格の 1 つとして Bluetooth がある．しかし，Bluetooth は通信可能範囲が限定されているため，宅外から宅内の機器を Bluetooth の仕組みで直接遠隔制御することができない．本論文では DTLS（Datagram Transport Layer Security）による暗号化トンネルを利用して Bluetooth の制御メッセージを遠隔地へ伝送することにより，遠隔地に存在する Bluetooth 搭載 IoT デバイスの遠隔制御システムを提案する．提案システムでは，遠隔地にある Bluetooth 搭載 IoT デバイスがつねに操作デバイスの近隣に存在しているように仮想的に認識するため，どこからでも遠隔制御を行うことができる．プロトタイプ実装により提案システムの動作検証を行った結果，モバイルインターネット環境を利用して宅内の Bluetooth 搭載 IoT デバイスを Bluetooth で規定されているタイムアウト時間より十分に短い範囲で通信できることを確認した．

キーワード：Bluetooth，IoT デバイス，仮想認識，遠隔制御

Evaluation of Remote Control System for IoT Devices with Bluetooth over DTLS

Mami Okada[1,†1,a)]　Hidekazu Suzuki[1,b)]

Received: September 30, 2017, Accepted: February 14, 2018

Abstract: A Bluetooth is one of short range wireless communication standard for controling Internet of things (IoT) devices. However, the Bluetooth range is limited, and a user typically cannot communicate with home devices from outside the home by means of the Bluetooth. In this paper, we propose a remote control system for IoT devices with Bluetooth existing in a remote place by transmitting Bluetooth control message over encrypted tunnel by datagram transport layer security (DTLS). In the proposed system, since remote IoT devices with Bluetooth are virtually recognized as always existing in the vicinity of the control device, remote control can be performed from anywhere. As a result of the operation verification of the proposed system by the prototype implementation, it was confirmed that using the mobile Internet environment, it is possible to communicate with IoT devices with Bluetooth in the home within a sufficiently shorter range than the timeout time specified by Bluetooth.

Keywords: Bluetooth, Internet of Things, virtual recognition, remote conrtol

1. はじめに

情報家電機器や電化製品をはじめとする我々の身の回りにあるあらゆるモノがインターネットにつながり，人々の生活をより便利にしたり，産業をより効率化したりすることが期待されている IoT（Internet of Things）が注目を集めている．文献 [1] によると，2021 年には 1 兆 4,000 億ドル規模の市場に拡大する予測が示されている．IoT は用途

[1] 名城大学大学院理工学研究科
Graduate School of Science and Technology, Meijo University, Nagoya, Aichi 468–8502, Japan
[†1] 現在，NEC ネッツエスアイ株式会社
Presently with NEC Networks & System Integration Corporation
[a)] mami.okada@ucl.meijo-u.ac.jp
[b)] hsuzuki@meijo-u.ac.jp

や対象により様々な通信規格が存在するが，ユーザのスマートフォンと連携することで様々なサービスを展開可能な短距離通信規格として，Bluetooth が主流となっている．

Bluetooth には初期に登場した BR（Basic Rate）の規格のほか，伝送速度を向上させた EDR（Enhanced Data Rate）の規格がよく採用されていた．現在は Bluetooth version 4.0 で新たに追加定義された LE（Low Energy）規格が主流となっており，従来の BR/EDR と比較して超低消費電力で通信できるという特徴を有している．Bluetooth プロトコルスタックはホストとコントローラから構成されており，両者の間に定義されている HCI（Host Controller Interface）層に含まれる HCI ドライバ（ソフトウェア）と HCI コントローラ（ハードウェアのファームウェア）間で HCI コマンド，HCI イベントと呼ばれる制御メッセージおよび HCI データを交換することにより，他の Bluetooth 機器との通信を実現している [2]．本論文では HCI 層で交換されるこれらの制御メッセージおよびデータをまとめて HCI メッセージと呼ぶ．

文献 [3] では，2016 年から 2022 年の間に IoT 市場向けの BLE チップセットの年間出荷台数は 524%増加することが予測されており，その中でもインダストリアルセンサ，スマートホームおよびビルオートメーションの分野は他の BLE IoT 市場よりも 3 倍のスピードで成長していくことが見込まれている．また文献 [4] では，ユーザの 86%以上がスマートホーム技術を利用する主な目的はエネルギーの管理および暖房や家電製品の遠隔制御であると回答している．以上から，今後，Bluetooth 搭載 IoT デバイスが宅内に普及し，それらを遠隔から制御したいというニーズは増加するものと考えられる．そこで本論文では，スマート家電やスマートホームに関連する宅内の Bluetooth 搭載 IoT デバイスを対象とした遠隔制御に焦点を当てて議論する．このような IoT デバイスは主に電源の ON/OFF，センシングデータや設定値などのテキストデータをやりとりするため，音声などリアルタイム性が要求されるアプリケーションおよびそれらが利用する Bluetooth 搭載 IoT デバイスは，本論文の制御対象から除外することとする．

Bluetooth は通信可能範囲が物理的に制限されているため，ユーザは宅内に設置された Bluetooth 機器を制御する場合は，その機器の近傍に位置していなければならない．そのため，Bluetooth 機器が宅内のホームゲートウェイなどに接続してインターネットに間接的につながっていたとしても，ユーザは宅外から宅内の機器を Bluetooth 対応アプリケーションで直接操作することができない．Bluetooth 4.2 ではスマートフォンなどとペアリングすることなく，ルータを介して直接インターネット接続することが可能な IPSP（Internet Protocol Support Profile）[5] と呼ばれるインターネット接続用プロファイルが定義されているが，Bluetooth 機器およびスマートフォンなどの操作機器にインストールされるソフトウェアがこのプロファイルを利用した仕様で開発されていなければならないこと，また宅内に設置するルータが 6LoWPAN（IPv6 over Low power Wireless Personal Area Networks）over BLE [6] をサポートしていなければならないなど，特定の条件が揃わなければ宅外から直接接続することはできない．

遠隔地に存在する Bluetooth 機器を操作するために，Bluetooth とは異なる別のプロトコルと連携することにより，Bluetooth 機器に制御メッセージを送信したり，通信結果を受信したりするサービスや技術が登場している [7], [8], [9], [10], [11]．さらに，Bluetooth 通信自体を遠隔地へ伝送することにより，ユーザは宅外先からでも宅内と同じ操作アプリケーションで Bluetooth 機器を操作できる手法も研究されている [12], [13]．しかし，従来の遠隔制御システムや既存研究では，ユーザは自身の位置に応じて操作アプリケーションを使い分けたり，専用の装置をつねに携帯したりしていなければならないなど，ユーザの利便性が良いとは必ずしもいえない．

そこで，筆者らはこれらの課題を解決すべく，ユーザビリティに優れた Bluetooth 機器の遠隔接続手法を提案してきた [14], [15], [16], [17]．文献 [14], [15] において，Bluetooth スタックでやりとりされる HCI メセージを NAT 配下に存在する宅内機器まで伝送するために，UDP トンネルを利用する仕様を提案し，文献 [16], [17] では BR/EDR に加えて BLE に対応させるための仕様拡張を行ってきた．しかし，遠隔制御を安全に行うために必要なセキュリティに関する検討が不足していた．

本論文では従来の提案システムに対して，ユーザの事前設定の負担が少なく，かつユーザ認証機能および暗号化通信機能を追加した遠隔制御システムを提案する．提案方式では，UDP トンネルで伝送されていた HCI メッセージを DTLS（Datagram Transport Layer Security）[18] による暗号化トンネルに変更して遠隔地に伝送することにより，ユーザは自身の位置を考慮することなく，通常の Bluetooth アプリケーションを利用して安全に遠隔制御および遠隔通信を行うことができる．提案システムのプロトタイプ実装，および実環境における性能評価を行うことにより，実用上問題ない遅延で通信できることを確認する．

以下，2 章で既存研究とその課題を示し，3 章で提案システムについて述べる．4 章で実装，5 章で評価について述べ，6 章でまとめる．

2. 既存研究

遠隔地に存在する Bluetooth 機器を制御する手段として，異なるプロトコルと連携して遠隔地にある専用の装置に制御メッセージを伝送し，遠隔地の Bluetooth 機器との通信結果を連携プロトコルにより取得する方法がある．東芝 HEMS（Home Energy Management System）[7] では，

東芝 Web サービスとして「フェミニティ倶楽部」を構築し，ユーザに様々なサービスを提供している．図 1 に示すように，外出先のユーザは HTTPS（Hypertext Transfer Protocol Secure）を用いて Web サービスにアクセスし，宅内に設置されているホームゲートウェイに宅内の Bluetooth 機器に対して制御命令を送信する．ホームゲートウェイは受信した命令に従って制御対象の機器と Bluetooth 通信を行い，制御結果を HTTPS により応答する．このほかにも PUCC（P2P Universal Computing Consortium）アーキテクチャ [8], [9], [10] や UbiGate [11] も第 3 のプロトコルを連携させることにより，同様の仕組みで遠隔地にある Bluetooth 機器を制御することができる．

しかし，ユーザは自身の位置に応じて操作アプリケーションを使い分ける必要がある．宅内にいる場合は Bluetooth で直接操作するアプリケーションを利用し，宅外では HTTPS で Web サービスにアクセスするアプリケーションを利用しなければならないため，ユーザビリティが高いとはいえない．また，文献 [4] において将来スマートホーム技術を利用することが期待されるユーザは，スマートホーム技術は信頼性が高く，使いやすく，かつ制御しやすいようにアプリケーションを設計する必要があると考えており，そのアプリケーションではユーザのプライバシ，データの機密性や安全なデータストレージを保証すべきであることが述べられている．これに対して，サーバを中継する方法は Bluetooth 機器の遠隔操作履歴がサーバ上に残ってしまう可能性があり，ユーザのプライバシが侵害される懸念がある．さらに，アプリケーション提供者は Web サービスと 2 種類のアプリケーションを開発し，さらに Web サービスを提供するサーバを継続的に運用しなければならず，遠隔制御を実現するためのコストが高いなどの課題がある．

ユーザの位置にかかわらず，つねに Bluetooth 通信を行う操作アプリケーションだけで遠隔地にある Bluetooth 機器を制御することが可能な手法として，図 2 に示す UbiPAN [13] が提案されている．この手法は UbiGate を拡張したものであり，遠隔地に存在する UbiGate Gateway（UGW）がその近隣に存在する Bluetooth 機器を探索し，その結果をユーザの近隣に設置した UGW まで SIP（Session Initiation Protocol）[19] を用いて伝送する．この状態でユーザが近隣の Bluetooth 機器を探索すると，近隣の UGW が遠隔地に存在している Bluetooth 機器の情報を返信することにより，遠隔地の Bluetooth 機器を発見することができる．物理的な距離の制約に縛られず，Bluetooth ネットワークが遠方まで拡大したように認識できることが，前述した既存サービスや既存研究にはない特徴である．このほかに，有線回線により Bluetooth ネットワークを拡張する手法 [12] も同様の考え方で遠隔制御できることを提案している．

しかし，ユーザはつねに UGW を携帯しなければならないことや，どの UGW 配下にどのようなサービスを実行できる Bluetooth 機器が存在するかなどの情報を収集および管理するための Register サーバを導入，運用する必要がある．

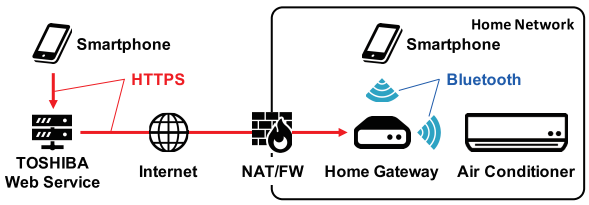

図 1　東芝 HEMS の概要
Fig. 1　Overview of Toshiba HEMS.

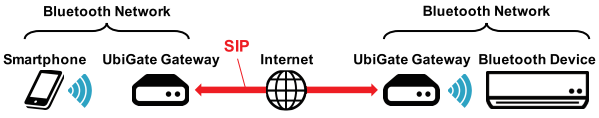

図 2　UbiPAN の概要
Fig. 2　Overview of UbiPAN.

3. 提案システム

3.1 概要

2 章で述べた既存研究の課題を解決するために，インターネット上のサーバを経由することなく，エンドツーエンドで宅内の Bluetooth 搭載 IoT デバイスを安全に遠隔制御できる技術が必要であると考えられる．そこで本論文では，Bluetooth ネットワークを仮想的に拡大するアプローチを採用した遠隔制御システムを提案する．図 3 に提案システムの概要を示す．提案システムは下記のデバイスから構成され，以後，Bluetooth 搭載 IoT デバイスを単に IoT デバイスと表記する．

- CD（Controle Device）：スマートフォンなどの操作端末
- ND（Neighbor Device）：CD の近隣に存在する Bluetooth 搭載 IoT デバイス
- BGW（Bluetooth Gateway）：宅内に設置する専用機器
- RD（Remote Device）：BGW の近隣に存在すし，CD

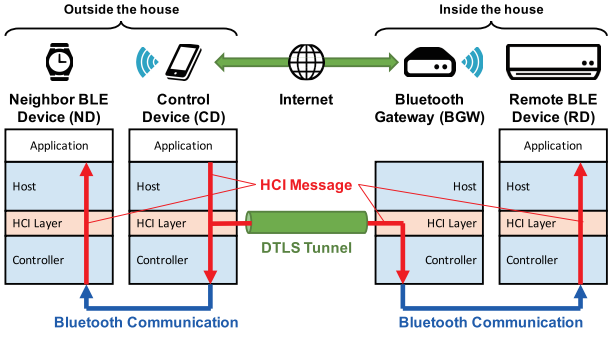

図 3　提案システムの概要
Fig. 3　Overview of the proposed system.

の遠隔制御対象となる宅内 Bluetooth 搭載 IoT デバイス

上記のうち，提案手法を実装する必要があるのは CD と BGW であり，ND および RD は市販されている通常の IoT デバイスである．

提案システムでは，他の既存技術と同様に宅内に BGW を設置するが，Bluetooth スタックに独自のモジュールを追加することにより，操作端末の近隣やインターネット上に専用のハードウェアが不要で，またプロトコル変換などの処理を行うことなく，さらにユーザは宅内と同じ通常の Bluetooth アプリケーションにより遠隔地の IoT デバイスを安全に制御することを実現する．

Bluetooth プロトコルスタックにおけるホストとコントローラの間で交換されている HCI メッセージのコピーを遠隔地に設置した BGW へ DTLS トンネル通信により伝送することで，BGW の Bluetooth コントローラに対して HCI メッセージを安全に届けることができる．これにより，CD は BGW の Bluetooth インタフェースを自身のインタフェースであるかのように操作することができる．BGW が近隣の IoT デバイスを探索または通信を行って RD からの応答を受信すると，Bluetooth コントローラからホストへ渡される HCI メッセージをフックし，自身の Bluetooth ホストではなく DTLS トンネル通信により CD 側へ送り返す．CD は ND だけでなく BGW から受け取る RD の情報を受け取ることができる．

これにより，既存研究のようにユーザは専用の装置を携帯する必要がなく，かつ場所の違いに影響されることなくつねに単一の操作アプリケーションで同じように ND と RD を同時に探索でき，ユーザが選択した IoT デバイスを操作することができる．なお，CD が ND と通信する場合は提案手法をいっさい利用することなく，通常の Bluetooth 通信の仕組みで制御する．

3.2　通信シーケンス

事前の準備として，ユーザはあらかじめ BGW との間で 3.2.1 項で述べるユーザ認証処理を行う．なお，自宅の NAT/ファイアウォール（FW）に対して宅外から BGW に対して DTLS 通信ができるよう，ポートフォワーディングの設定を行っているものとする．

3.2.1　ユーザ認証フェーズ

提案システムでは CD と BGW の双方向認証を行うため，公開鍵証明書およびパスワードを用いた 2 種類のユーザ認証モードをサポートする．なお，以下に示すユーザ認証処理はユーザが宅外のネットワークに移動した際に 1 回だけ実施するものであり，宅内に存在する場合は必要としない．

(1) 公開鍵証明書によるユーザ認証モード

この認証モードでは，BGW において CA（Certification Authority）を構築し，BGW 用にサーバ証明書を，CD 用にクライアント証明書を発行する．**図 4** に公開鍵証明書を用いたユーザ認証シーケンスを示す．CD が宅外のネットワークに接続すると，BGW に対して DTLS ハンドシェイクを開始し，両者は自身の公開鍵証明書を交換して共通鍵を生成する．CD および BGW は DTLS の枠組み内で双方向認証が行われ，DTLS ハンドシェイクが完了するとユーザ認証も完了し，DTLS トンネルが生成される．

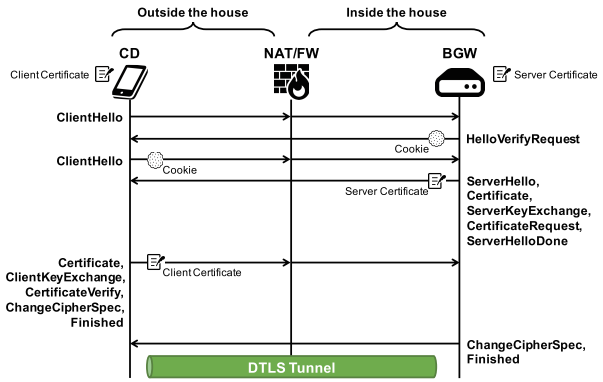

図 4　公開鍵証明書を用いたユーザ認証シーケンス

Fig. 4　Sequence of the user authentication using public key certificates.

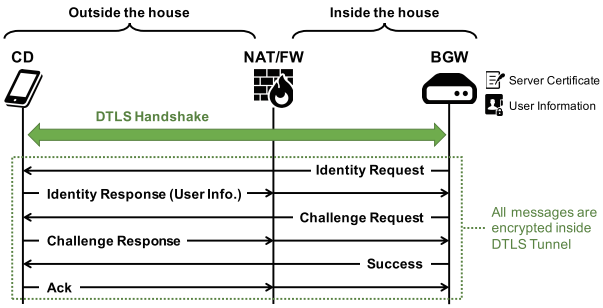

図 5　パスワードを用いたユーザ認証シーケンス

Fig. 5　Sequence of the user authentication using the password.

(2) パスワードによるユーザ認証モード

公開鍵証明書を用いたユーザ認証モードでは，CD に対してクライアント証明書を発行してインストールする作業が必要となり，ユーザにとって負担となることが考えられる．そこで，CD にクライアント証明書を発行したり，インストールしたりする手間を省略する代わりに，ユーザ名とパスワードを用いて CD を認証する方法を選択することができる．この認証モードでは，BGW にはサーバ証明書のほか，ユーザ名とパスワードを記載したユーザ認証情報を準備しておく．

図 5 にパスワードを用いたユーザ認証シーケンスを示す．まず，CD が BGW に対して DTLS ハンドシェイクすることは変わりないが，DTLS ハンドシェイクでは BGW 側のみサーバ証明書により認証することができる．そのため，DTLS ハンドシェイクが完了して

DTLS トンネルが構築された後，クライアント認証処理に移る．

まず，BGW が CD に対して Identity Request メッセージを送信し，ユーザ認証を要求する．CD はユーザ名とパスワードを入力し，Identity Response メッセージにより返信する．さらに再送攻撃によるなりすまし行為を防止するために，BGW は CHAP（Challenge-Handshake Authentication Protocol）[20] で採用されているチャレンジを CD へ行い，CD はそれに対する応答を返す．以上の処理により，CD と BGW 間で DTLS による暗号化トンネルが形成され，かつ双方向認証が完了する．

なお，この双方向認証手順については，EAP-TTLSv0（Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0）[21] および PEAPv0（Protected EAP）[22] などのユーザ ID とパスワードを用いた認証プロトコルに準じて定義したものである．

3.2.2　Bluetooth 通信フェーズ（機器探索）

3.2.1 項で示したユーザ認証フェーズの完了後，CD で Bluetooth 通信を行う場合は**図 6** に示す流れで Bluetooth 通信が行われる．まず，CD 側で Bluetooth 通信を行う操作アプリケーションを起動して近隣の Bluetooth 機器の探索を開始すると，Bluetooth ホストからコントローラに向けて HCI メッセージが渡される．ここで，HCI 層において HCI メッセージを複製し，元の HCI メッセージはそのままコントローラに渡して，CD は自身の Bluetooth インタフェースを用いて近隣の IoT デバイス ND を探索する．ND を発見した場合は通常の Bluetooth プロトコルスタックの処理手順により探索結果がアプリケーションに渡される．

一方，複製された HCI イベントメッセージはユーザ認証フェーズで構築した DTLS トンネルを利用して BGW へ伝送される．BGW は CD から HCI メッセージを受信すると，自身の Bluetooth コントローラへ HCI メッセージを渡し，BGW の近隣に存在する IoT デバイス機器 RD を探索する．RD を発見した場合は逆の手順により HCI メッセージを CD 側へ伝送する．CD は BGW から受信した HCI メッセージを自身の Bluetooth ホストへ渡す．以上の処理により，操作アプリケーションに RD の情報も渡される．

3.2.3　Bluetooth 通信フェーズ（データ通信）

ユーザが探索結果より操作したい IoT デバイスを選択すると，その IoT デバイスの BD アドレスを宛先とした HCI メッセージがホストからコントローラに向かって渡される．HCI 層では HCI メッセージに記載された宛先 BD アドレスを用いて，3.2.2 項で作成したトンネルテーブルを検索する．宛先が RD の場合は該当するエントリが見つかるため，**図 7** に示すとおり，HCI メッセージを複製せずにフックし，DTLS トンネルを利用して BGW へ伝送する．以後は機器探索時と同じ手順で HCI メッセージを処理することにより，CD と RD 間のデータ通信は Bluetooth over DTLS により行われる．

一方，宛先が ND の場合，トンネルテーブルに該当するエントリが存在しないため，HCI メッセージをフックせずに自身のコントローラへ渡す．これにより，CD と ND 間のデータ通信は通常の Bluetooth と同じ手続きにより行われる．

以上により，ユーザは自身の場所や Bluetooth で定められている通信可能範囲を考慮することなく，また制御対象となる IoT デバイスの位置の違いを気にすることなくシームレスに接続することができる．

4.　実装

提案システムを実現するためには，Bluetooth プロトコルスタックにおける HCI 層で HCI メッセージの複製を BGW へ伝送し，BGW 側から受け取った HCI メッセージを CD のプロトコルスタックに戻す処理が必要である．そこで，提案システムを実現するために，Linux PC を利用し，Linux に実装されている Bluetooth プロトコルスタック BlueZ [23] のカーネルモジュールを拡張した．また，DTLS トンネルの構築，ユーザ認証処理および HCI メッセージの伝送を行うために HCI Forwarder デーモンを実装した．**図 8** にモジュール構成を示し，その詳細を以下に説明する．

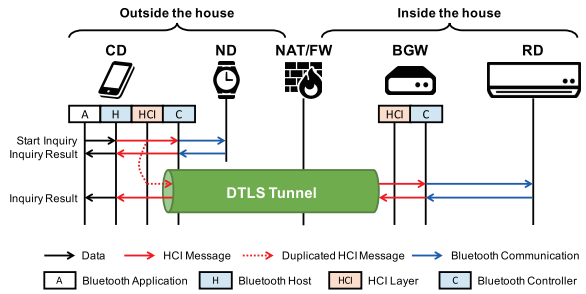

図 6　Bluetooth デバイス探索シーケンス

Fig. 6　Sequence of the Bluetooth device discovery.

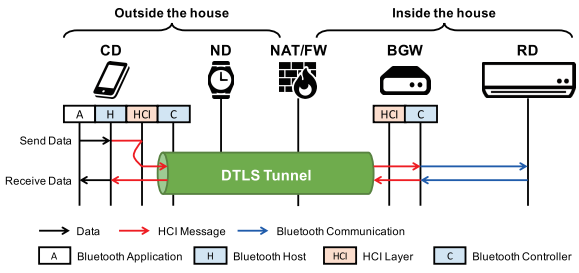

図 7　RD との Bluetooth データ通信シーケンス

Fig. 7　Sequence of the Bluetooth data communication with RD.

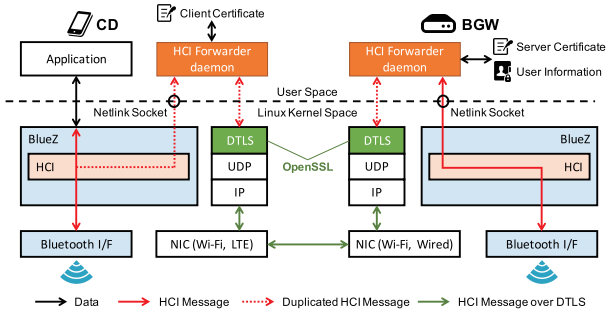

図 8　モジュール構成

Fig. 8　Module structure.

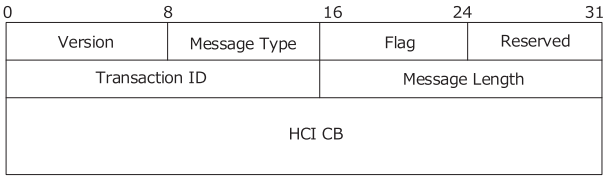

図 9　独自ヘッダのフォーマット

Fig. 9　Original header format.

4.1　Bluetooth プロトコルスタックの拡張

4.2　HCI Forwarder デーモン

CD 側の BlueZ に実装されている HCI 層における HCI メッセージの送信処理部に HCI メッセージが格納されているソケットバッファを複製する処理を追加した．複製したソケットバッファは Bluetooth カーネルモジュールに新たに追加した関数へ渡し，**図 9** に示す独自ヘッダを複製したソケットバッファの先頭に付与する．独自ヘッダの各フィールドの定義は以下のとおりである．

- Version：独自ヘッダ付 HCI メッセージのバージョン情報．
- Message Type：HCI メッセージの種類．
- Flag：HCI メッセージの正常/異常を示すフラグ．
- Reserved：予約用フィールド．
- Transaction ID：トランザクションを示す識別子．
- Message Length：独自ヘッダ以降のメッセージ長．
- HCI Control Block：HCI メッセージの処理に必要な情報．

独自ヘッダが付与された HCI メッセージは Netlink ソケット [24] を用いてユーザランドで動作している HCI Forwarder デーモンへ渡される．また，BGW 側の HCI 層における HCI メッセージの送信処理部では，HCI Forwarder デーモンから受け取った CD の HCI メッセージを BlueZ のコントローラへ渡す処理を追加した．

一方，BGW 側の HCI 層における HCI メッセージ受信処理部では，RD から受信した HCI イベントメッセージを Netlink ソケットにより HCI Forwarder デーモンへ渡すようにした．CD 側の HCI メッセージ受信処理部では，HCI Forwarder デーモンから受け取った CD の HCI メッセージを BlueZ のホストへ渡す処理を追加した．この処理により，CD は BGW から独自ヘッダ付きの HCI イベントメッセージを受け取ると，独自ヘッダに記載されている情報をもとに HCI イベントメッセージを生成し，CD 自身の BlueZ を通じ既存の Bluetooth Application へ渡される．これらの処理により，遠隔地から受信した HCI メッセージを，自端末の Bluetooth Application と Bluetooth I/F 間で交換される HCI メッセージと同じものとして処理できるように工夫した．

HCI Forwarder デーモンは CD と BGW においてバックグラウンドで動作し，拡張した Bluetooth カーネルモジュールとのメッセージ交換を行う Netlink ソケットおよび DTLS トンネルを用いたメッセージ送受信を行うためのデータグラムソケットを作成する．DTLS ハンドシェイクおよび暗号化や認証処理を行うために，OpenSSL ライブラリ（version 1.0.2g）[25] を利用した．なお，DTLS のプロトコルバージョンは 1.2 を採用した．

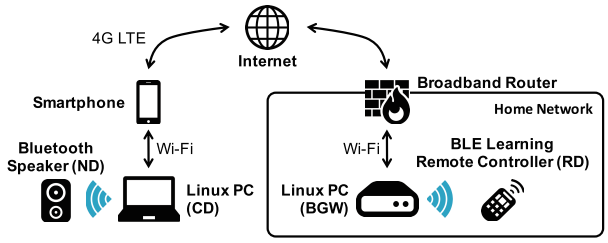

図 10　実験環境

Fig. 10　Experimental environment.

5.　評価

5.1　実験方法

図 10 に示す環境において，プロトタイプ実装した提案システムの動作検証および性能評価を行った．Ubuntu 17.04 をインストールしたラップトップ PC を CD および BGW とし，Bluetooth4.0+EDR/LE 対応 USB アダプタ（BUFFALO 社製 BSBT4D09BK [*1]）を装着した．CD はスマートフォンをテザリングして 4G LTE 回線によりインターネットに接続し，BGW は研究室の LAN に設置してブロードバンドルータ（BBR）を通じてインターネットに接続した．CD の近隣には EDR 対応の Bluetooth スピーカ（日本電話施設社製 OmniBeat [*2]）を ND として，また BGW の近隣には BLE 対応の学習リモコンユニット（ラトックシステム社製 REX-BTIREX1 [*3]）を RD としてそれぞれ配置し，CD から直接 RD を探索できない位置関係とした．

上記の環境において，公開鍵証明書を用いたユーザ認証

[*1] http://buffalo.jp/product/peripheral/wireless-adapter/bsbt4d09bk/
[*2] http://www.nds-g.co.jp/files/news/pdf/4a5a826e.pdf
[*3] http://www.ratocsystems.com/products/subpage/btirex1.html

表 1　各機器間の平均 RTT
Table 1　Average round-trip times between each device.

	CD–BBR	BBR–BGW	BGW–RD
RTT [ms]	81.78	0.88	35.80

処理を行った後，CD 上で Bluetooth 機器を探索するコマンド `hcitool inq` を実行し，ND および RD を探索できるか確認した．その際，Wireshark を用いてユーザ認証時間を，またコマンド `hcidump` を用いて機器探索する際の HCI メッセージのやりとりをダンプし，Wireshark を用いて CD が最初に HCI コマンドを発行してから ND および RD の情報が記載された最初の HCI イベントを受け取るまでに要した時間を計測した．なお，提案システムを適用しない状態でコマンド `ping` およびコマンド `l2ping` [*4] を用いて事前に 100 回測定した各機器間の平均 RTT（Round-Trip Time）は**表 1** のとおりである．また，公開鍵証明書の RSA 公開鍵長は 2,048 bit，ダイジェストアルゴリズムは SHA-256 として生成したものを用いた．

5.2　測定結果

表 2 に提案システム適用時におけるユーザ認証時間と機器探索時間を 10 回測定した結果を示す．提案方式における CD および BGW 間で行われたユーザ認証処理は 445.71 ミリ秒で完了しており，ユーザが宅外のネットワークに接続したときに 1 回だけ発生する処理であることを考えると，実用上問題にならない．また，ND および RD の探索時間はそれぞれで 685.49 ミリ秒，489.92 ミリ秒であった．別途，同一環境において提案システムを適用しない通常時の ND 探索時間を測定したところ，589.18 ミリ秒であったことをふまえると，提案システムでは ND 探索時間が増加していることが分かる．これは提案方式において HCI メッセージを複製した際，先に HCI Forwarder デーモンに複製した HCI メッセージを渡してから，元の HCI メッセージを Bluetooth コントローラへ渡す順序で実装を行っていることが原因であると考えられる．一方，RD の探索には ND の探索時間に加えて CD と BGW 間の通信遅延および DTLS に基づく暗号化および認証処理時間が加算されているが，Bluetooth の仕様で定義されている探索間隔 2.56 秒の範囲で収まっており，実用上問題ないと考えられる．

次に，機器探索時の実測値などからデータ通信に関わる評価を行った．Bluetooth の仕様書には，BR/EDR のタイムアウト値は 5 秒と定義されている．データ通信時においても機器探索時と同様に HCI メッセージをフック，カプセル化して伝送する．機器探索時の HCI メッセージのフックおよびカプセル化にかかる時間は平均 0.26 ミリ秒であった．通常の Bluetooth 通信にかかる時間および LTE 通信にかかる時間は，表 2 よりそれぞれ 35.80 ミリ秒，81.78 ミリ秒であった．結果として，データ通信に要する時間は平均 117.84 ミリ秒と算出することができる．したがって，LTE ネットワークが輻輳して遅延が増加したと想定しても，Bluetooth のタイムアウト値より十分に短いことから，アプリケーションに対する伝送遅延の影響はほとんどないと思われる．

[*4] Bluetooth の L2CAP（Logical Link Control and Adaptation Protocol）を用いて Echo Request/Response の交換に要した時間を測定するコマンド．

表 2　測定結果
Table 2　Measurement results.

	Min [ms]	Avg [ms]	Max [ms]
User authentication	255.16	332.74	445.71
ND discovery	122.99	685.49	974.40
RD discovery	102.79	489.92	907.39

5.3　考察

本論文では BlueZ を利用して提案システムのプロトタイプを実装したが，今後は Android スマートフォンへの移植を検討している．Android は Linux カーネルを採用しており，バージョン 4.1 までは PC 向け Linux と同様に BlueZ が採用されていた．そのため，提案システムの機能をクロスコンパイルすることにより，Android スマートフォンへ移植することができる．ただし，現在市販されている Android スマートフォン（バージョン 4.2〜5.1）は “Bluedroid” と呼ぶ新しい Bluetooth プロトコルスタックに変更されている．さらに Bluedroid は “Fluoride” と名称を変更し，Android 6.0 以降に実装されている [26]．

そこで，提案方式が Fluoride に適用できるかを検討した．**図 11** に Fluoride の Bluetooth プロトコルスタックを示す [27]．Fluoride では HCI 層や HCI メッセージをトレースするためのカスタムエクステンションを追加するために，ベンダ拡張が可能な仕組みが用意されている．そのため，`libbt-hci` モジュールを開発することにより，提案システムの機能を実装できると考えられる．したがって，最新の Android スマートフォンを CD として利用して，提案システムを実現することができる．

6.　まとめ

本論文では，遠隔地にある Bluetooth 搭載 IoT デバイスを仮想的に発見，接続および通信するシステムを提案した．提案システムでは Bluetooth プロトコルスタックにおけるホストとコントローラ間でやりとりされる制御メッセージを DTLS トンネルを用いて遠隔地の BGW に伝送することにより，BGW の近傍に存在する RD を CD の近隣に存在するかのように認識させることができることを示した．また，提案システムのプロトタイプ実装を行い，実環境において動作検証および通信遅延を評価した．その結果，

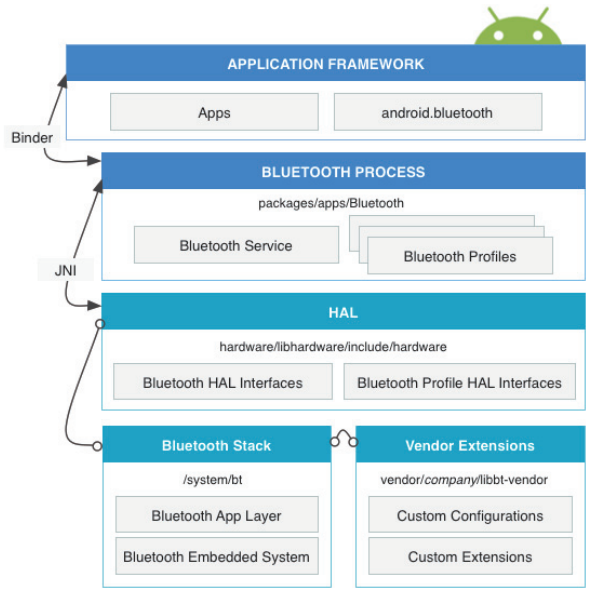

図 11　Fluoride の Bluetooth プロトコルスタック
Fig. 11　Bluetooth protocol stack of Fluoride.

Bluetooth で規定されているタイムアウト時間内に CD が遠隔地の RD を発見できることを確認した．また，現在市販されている Android スマートフォンに提案手法を適用できることを示した．

今後は提案システムを Android へ移植し，スマートフォンでの実現を目指す．また，提案システムの応用として，Bluetooth の制御メッセージだけでなく，様々なデバイスの制御メッセージを遠隔地に設置したゲートウェイに送信することにより，ゲートウェイに装着されているデバイスをスマートフォンに仮想的に装着して遠隔制御する手法について検討する予定である．

謝辞　本研究の一部は，東北大学電気通信研究所における共同プロジェクト研究の支援によって行われた．

参考文献

[1] Shirer, M. and Torchia, M.: Worldwide Spending on the Internet of Things Forecast to Reach Nearly $1.4 Trillion in 2021, According to New IDC Spending Guide, IDC Research, Inc. (online), available from ⟨http://www.idc.com/getdoc.jsp?containerId=prUS42799917⟩ (accessed 2017-07-29).
[2] Bluetooth SIG: BLUETOOTH SPECIFICATION Version 4.0, Technical Report 1, Bluetooth SIG (2010).
[3] Hatler, M., Gurganious, D. and Kreegar, J.: Bluetooth Low Energy IoT: A Market Dynamics Report, Technical report, ON World Inc. (2017).
[4] Wilson, C., Hargreaves, T. and Hauxwell-Baldwin, R.: Benefits and risks of smart home technologies, *Energy Policy*, Vol.103, pp.72–83 (2017).
[5] Internet WG: Internet Protocol Support Profile Bluetooth Specification, Technical report, Bluetooth SIG (2014).
[6] Nieminen, J., Savolainen, T., Isomaki, M., Patil, B., Shelby, Z. and Gomez, C.: IPv6 over BLUETOOTH(R) Low Energy, RFC 7668, IETF (2015).
[7] 一色正男，河口俊朗，平原茂利夫：広がる東芝ネットワーク家電 “フェミニティ” シリーズ，東芝レビュー，Vol.60, No.4, pp.3–27（オンライン），入手先 ⟨https://www.toshiba.co.jp/tech/review/2005/04/60_04pdf/a07.pdf⟩ (2005).
[8] Sumino, H., Ishikawa, N., Murakami, S., Kato, T. and Hjelm, J.: PUCC Architecture, Protocols and Applications, *Proc. 4th IEEE Consumer Communications and Networking Conference, CCNC 2007*, pp.788–792 (2007).
[9] 伊藤崇洋，加藤悠一郎，峰野博史，石川憲洋，水野忠則：異種デバイス連携基盤を用いたセンサ・家電制御アプリケーション，情報処理学会研究報告コンピュータセキュリティ（CSEC），Vol.2011-CSEC-52, No.35, pp.1–6 (2011).
[10] 田中　剛，伊藤崇洋，加藤悠一郎，峰野博史，水野忠則：Android 端末を用いた異種ネットワークデバイス連携システムの開発，マルチメディア，分散協調とモバイルシンポジウム 2011 論文集，DICOMO2011, Vol.2011, pp.1257–1264 (2011).
[11] Bissyandé, T.F.D., Réveillère, L. and Bromberg, Y.-D.: UbiGate: A Gateway to Transform Discovery Information into Presence Information, *Proc. 4th International Workshop on Services Integration in Pervasive Environments, SIPE 2009*, pp.19–24 (2009).
[12] 井波政朗，丹　康雄：Bluetooth ネットワークの有線拡張方式に関する検討，電子情報通信学会技術研究報告，CS2003, Vol.103, No.415, pp.47–52 (2003).
[13] Albert, J., Bissyandé, T.F., Bromberg, Y.-D., Chaumette, S. and Réveillère, L.: UbiPAN: A Bluetooth Extended Personal Area Network, *Proc. 4th International Conference on Complex, Intelligent and Software Intensive Systems, CISIS 2010*, pp.774–778 (2010).
[14] 津田一磨，鈴木秀和，旭　健作，渡邊　晃：遠隔地にある Bluetooth 機器間のシームレス接続手法の実装，マルチメディア，分散，協調とモバイル（DICOMO2013）シンポジウム論文集，Vol.2013, No.1, pp.805–811 (2013).
[15] Tsuda, K., Suzuki, H., Asahi, K. and Watanabe, A.: Proposal for a Seamless Connection Method for Remotely Located Bluetooth Device, *Proc. 7th International Conference on Mobile Computing and Ubiquitous Networking, ICMU 2014*, pp.78–79 (2014).
[16] 岡田真実，鈴木秀和：遠隔地にある Bluetooth LE 機器のシームレス接続手法の検証，情報処理学会研究報告コンシューマ・デバイス&システム（CDS），Vol.2016-CDS-17, No.13, pp.1–7 (2016).
[17] Okada, M. and Suzuki, H.: Implementation of Seamless Connection System for Bluetooth Low Energy Devices in Remote Locations, *Proc. 35th International Conference on Consumer Electronics, ICCE 2017*, pp.341–342 (2017).
[18] Rescorla, E. and Modadugu, N.: Datagram Transport Layer Security Version 1.2, RFC 6347, IETF (2012).
[19] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and Schooler, E.: SIP: Session Initiation Protocol, RFC 3261, IETF (2002).
[20] Simpson, W.: PPP Challenge Handshake Authentication Protocol (CHAP), RFC 1994, IETF (1996).
[21] Funk, P. and Blake-Wilson, S.: Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0 (EAP-TTLSv0), RFC 5281, IETF (2008).
[22] Kamath, V., Palekar, A. and Wodrich, M.: Microsoft's PEAP version 0 (Implementation in Windows XP SP1), Internet-draft, IETF (2002). draft-kamath-pppext-peapv0-00.txt.

[23] BlueZ Project: BlueZ, available from ⟨http://www.bluez.org/⟩ (accessed 2016-07-26).
[24] Salim, J., Khosravi, H., Kleen, A. and Kuznetsov, A.: Linux Netlink as an IP Services Protocol, RFC 3549, IETF (2003).
[25] OpenSSL Project: Cryptography and SSL/TLS Toolkit, available from ⟨https://www.openssl.org/⟩ (accessed 2017-07-29).
[26] Git at Google: Fluoride Bluetooth stack, available from ⟨https://android.googlesource.com/platform/system/bt/⟩ (accessed 2017-07-29).
[27] Android Open Source Project: Bluetooth, available from ⟨https://source.android.com/devices/bluetooth.html⟩ (accessed 2016-07-29).

岡田 真実　（学生会員）

2015 年名城大学理工学部情報工学科卒業．2018 年 3 月同大学大学院理工学研究科情報工学専攻修士課程修了．2018 年 4 月 NEC ネッツエスアイ株式会社に入社．在学時代は主としてホームネットワークに関する研究に従事．修士（工学）．

鈴木 秀和　（正会員）

2004 年名城大学理工学部情報科学科卒業．2009 年同大学大学院理工学研究科電気電子・情報・材料工学専攻博士後期課程修了．2008 年日本学術振興会特別研究員．2010 年名城大学理工学部助教．2015 年より同大学理工学部准教授および東北大学電気通信研究所共同研究員を兼任．ネットワークセキュリティ，モバイルネットワーク，ホームネットワーク等の研究に従事．博士（工学）．IEEE，ACM，WCTRS，電子情報通信学会各会員．