Vol.2017-DPS-171 No.30
Vol.2017-MBL-83 No.30
Vol.2017-ITS-69 No.30
2017/6/2

ディジタル証明書認証を活用する移動透過通信技術の開発

宮崎 祐哉[1,a)]　内藤 克浩[1]　鈴木 秀和[2]　渡邊 晃[2]

概要：スマートフォンや Internet of Things(IoT) デバイスは現代の生活の中で欠かせないものになりつつある．スマートフォンでは, 映像配信アプリケーションやメッセージアプリケーションなど様々なアプリケーションが開発されてきている．また, ドアの開閉を電子的に管理するスマートロック, スマートフォンから家電を制御できる仕組みなど,IoT 製品も目覚ましく普及してきている．しかし, これらのサービスは Network Address Port Translation(NAPT) ルータによりデバイス間での直接通信が難しいことが知られている．上記課題を解決するため, 多くのサービスでクライアントサーバモデルを採用している．しかし, クライアントサーバモデルを採用した場合, デバイス間の通信は外部のサーバを経由するスター型通信となり, 冗長な経路となる課題がある．また, サーバを経由するため通信時のログが第三者に渡る可能性も懸念されている．著者らは, デバイス間の直接通信を実現するプロトコルである Network Traversal with Mobility(NTMobile) の開発を進めてきた．NTMobile では, デバイス間の直接通信機能と独自手法のセキュリティを実装しており, 安全な通信をエンド間で実現可能である．一方, デバイスの認証はサーバーに対してのみ行われているため, デバイス自身が相互に認証を行う手段は提供されていない．また,NTMobile を構成する装置が攻撃された場合, 理論上暗号鍵を入手可能となる可能性がある．本論文では,NTMobile のセキュリティ技術の向上を目的として Public Key Infrastructure(PKI) の技術を導入する．提案手法では, ディジタル証明書を利用することにより, より安全に暗号鍵をデバイス間で共有可能となるとともに, 相手端末の認証が可能になる．実証実験では, 通信開始時の処理遅延に着目し, 既存の NTMobile の実装と比較することにより有効性を確認する．

キーワード：移動透過性, 通信接続性, PKI, NTMobile, ディジタル証明書

1. はじめに

近年, スマートフォンや Internet of Things(IoT) デバイスをはじめとした, インターネットへ接続可能なデバイス数が増加している．現在, インターネットの基盤技術には Internet Protocol(IP) が採用されており, 主に IPv4 が利用されている．IPv4 アドレスは 32 ビットで表現されることから, 近年の接続デバイス数の増加に伴い,IPv4 アドレスの枯渇が問題となっている．そのため, グローバルアドレスとプライベートアドレスの変換を行う Network Address Port Translation(NAPT) ルータや Internet Protocol version 6(IPv6) の導入が行われてきた [1]．しかし,NAPT ルータはインターネット側からの通信を遮断すると共に,IPv4 と IPv6 間には互換性がないことから, 通信接続性の課題が顕著になりつつある [2], [3]．デバイス間通信を想定した場合, これらの課題はクライアントサーバモデルを採用することで解決可能である．そのため, 多くの既存サービスにおいてクライアントサーバモデルが採用されている．しかし, クライアントサーバモデルでは必ずサーバを経由する冗長な経路となる問題がある上, サーバに通信内容を記録されるおそれがある．先行研究では, クライアントサーバモデルを用いないエンド間通信により, 通信接続性を確保する研究が行われている [4], [5], [6]．エンド間通信により, ネットワークトラヒックの軽減や通信速度の向上など様々な利点が見込まれる．

現代ではスマートフォンをはじめとした多くのデバイスが, インターネットに接続するための複数の無線通信技術を実装している．そのため, 品質の良い通信インターフェースを選択し, 回線を切り替えることにより, 通信品質の向上を図る試みが始められている．しかし, インターフェースの切り替えに伴いネットワークの位置情報が変化するため,IP アドレスが変化し, 通信が切断される課題がある．これは,IP アドレスが位置情報と識別子の二つの役割を持つ

[1] 愛知工業大学情報科学部
Faculty of Information Science, Aichi Institute of Technology, Toyota, Aichi 470-0392, Japan
[2] 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
[a)] miyazaki121@pluslab.org

ためである. 先行研究では位置情報と識別子を分離する手法が提案されており, 位置情報の変化時にも通信を継続可能な技術として知られている [7]. この技術は一般に移動透過技術と呼ばれ様々な研究が行われている [8], [9], [10].

通信接続性と移動透過性それぞれを実現する技術は提案されているが, 両者を一挙に実現可能な技術は十分に開発されていなかった. 著者らは, 現代ネットワークで通信接続性と移動透過性の両者を解決するため,Network Traversal with Mobility(NTMobile) の開発を進めてきた [11], [12]. NTMobile では, 仮想 IP アドレスに基づく通信を提供することにより移動透過性を実現するほか,Direction Coordinator(DC) と呼ばれる装置による適切な経路指示により通信接続性を実現している. NTMobile を導入した端末は, ネットワークの位置情報に関わらずお互いに通信を開始でき, 移動時にも通信が切断されなくなる. また,NTMobile を導入した端末間では暗号鍵を共有するため, 通信の秘匿性が担保される. しかし, 暗号鍵の共有には外部装置である DC が関与するため暗号鍵が流出する恐れがある. 2013 年にはアメリカ国家安全保障局 (National Security Agency: NSA) が,PRISM と呼ばれる盗聴システムを用いて大手サービスサーバを盗聴していたことが告発された [13]. この事件のように, ユーザーがセキュリティを意識している場合にも外的要因で秘匿性が損なわれる恐れがある. また,NTMobile を導入した端末は相手端末を認証する手段を備えておらず, 正否の判断は DC に依存している. IoT 推進コンソーシアムの発行するセキュリティガイドライン ver1.0 によると,IoT デバイスは不特定多数のデバイスと通信するおそれがあるため, 相手端末の認証が必要であると述べている [14].

本研究では NTMobile に Public Key Infrastructure(PKI) を導入しセキュリティの向上を図る. NTMobile 端末にディジタル証明書を発行し, より安全な暗号鍵の共有方法と相手端末の認証方法を提供する. 本研究では,Linux 上で既存の NTMobile の拡張実装を行い, 提案方式の有効性及び性能を評価する.

2. NTMobile の概要

図 1 に NTMobile のシステムモデルを示す. NTMobile は,NTMobile を搭載した端末である NTM 端末,NTM 端末を管理する DC,NTM 端末の認証を行う Account Server(AS), 通信を中継する Relay Server(RS) から成り立つ. NTMobile では移動透過性を実現するため,DC が NTM 端末へ Fully Qualified Domain Name(FQDN) と仮想 IP アドレスを割り当てる. 通信開始時は, 相手 NTM 端末の FQDN の名前解決をトリガとして NTMobile の処理が実行される. この処理により, 相手 NTM 端末の実 IP アドレスと仮想 IP アドレスが入手できる. 得られた仮想 IP アドレスを名前解決の結果としてアプリケーションへ通知し, 通信相手の識別子として利用する. アプリケーションは仮想 IP アドレスに基づいた通信を行うため,NTM 端末が移動した場合にもアドレス変化の影響を受けない. アプリケーションから生成されるメッセージは仮想 IP アドレス宛となるため,NTMobile のカプセリング処理機能が実 IP アドレスでカプセル化を行う. カプセル化パケットは User Datagram Protocol(UDP) トンネルを経由することで, 相手 NTM 端末へ送信される. 受信パケットのデカプセル化もカプセリング処理機能により行われ, 仮想 IP アドレス宛のパケットとしてアプリケーションへ渡す処理が行われる. 以下に構成要素の詳細を示す.

- Direction Coordinator (DC)
  経路構築指示や仮想 IP アドレスの配布を行う装置である. DC は NTM 端末の FQDN, 実 IP アドレス, 仮想 IP アドレス,NAPT の実 IP アドレス, ポート番号を記録する. 通信接続性を実現するため, 登録されたアドレス情報に基づき適切に経路の構築指示を行う. 各 DC は仮想 IP アドレス用のアドレスプールが割り当てられており, 重複がないように NTM 端末へ割り当てる. DC はデュアルスタックネットワーク上に分散配置されることを想定している.
- Account Server (AS)
  NTM 端末の認証と DC の割り当てを行う装置である. 事前の登録情報に基づき認証を行い, 認証成功時には DC の割り当てを行う. 同時に, 割り当てた DC と NTM 端末間での通信で用いる共通鍵を両者に配布する. AS は, デュアルスタックネットワーク上に配置される.
- Relay Server (RS)
  NTM 端末間で直接通信が行えない場合に通信を中継する装置である. NTMobile は,IPv4/IPv6 ネットワークに対応している. 両プロトコル間では互換性がないため,RS をデュアルスタックネットワーク上に配置しパケットを中継させる. また, 両者が NAPT 配下にいる場合にも導入される. RS の有無は DC の指示により決定されるため,DC 同様に分散配置が可能である. RS はデュアルスタックネットワーク上に配置される.
- NTM 端末
  NTMobile を搭載した端末を指す. NTM 端末は初回に AS と通信を行い認証を行うことにより DC が割り当てられる. 割り当て処理により,NTM 端末は管理される DC のホストとしての FQDN を得る. その後, 割り当てられた DC へ自身の位置情報を登録する. NTM 端末との通信時には DC へ経路構築の依頼を行う. 経路構築時に相手 NTM 端末と暗号鍵を共有するため, 安全なトンネル通信が可能である.

Vol.2017-DPS-171 No.30
Vol.2017-MBL-83 No.30
Vol.2017-ITS-69 No.30
2017/6/2

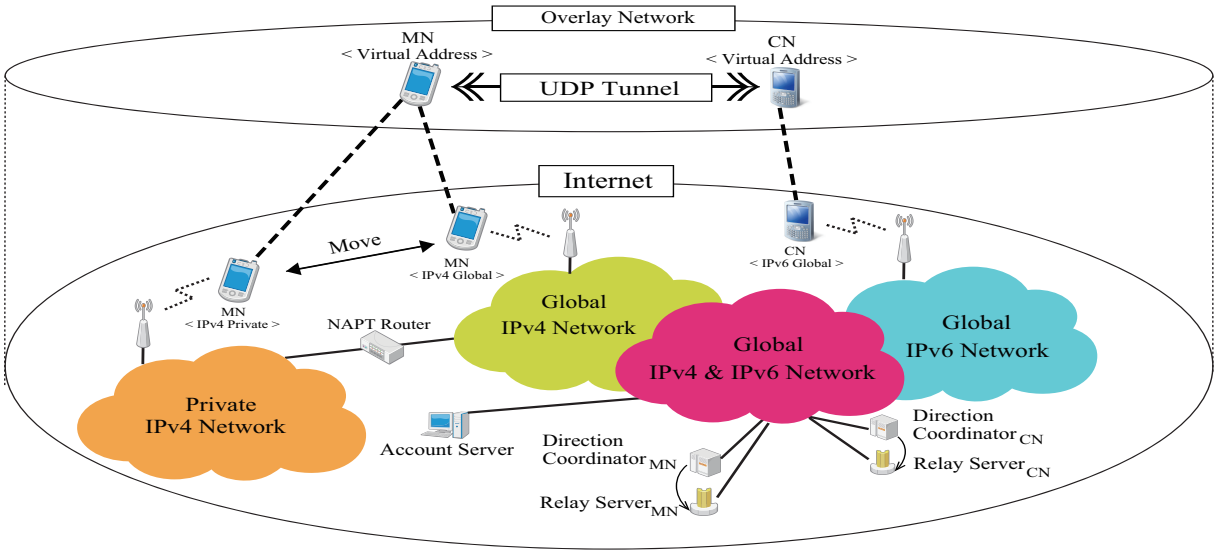

図 1　NTMobile のシステムモデル

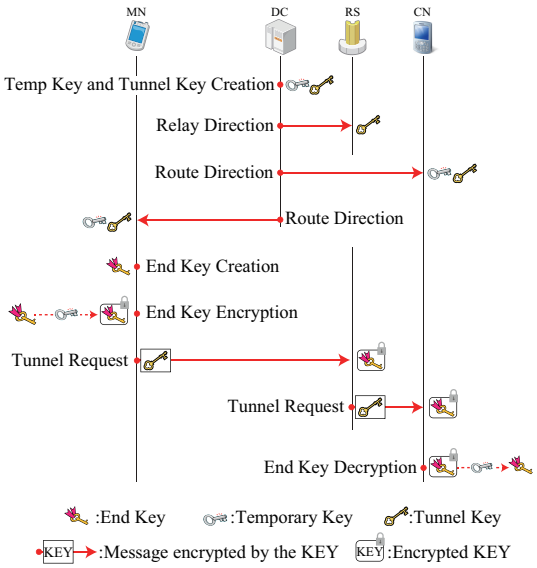

図 2　NTMobile におけるエンド鍵の共有手法

2.1　NTMobile における暗号鍵の共有手法

図 2 は経路構築時における暗号鍵の共有手法の概要を示す．NTMobile ではエンド端末間で暗号鍵を共有する独自のセキュリティ手法を用いており, エンド端末間の通信の安全性を実現している．ここで,NTMobile の AS/DC/RS 間は Secure Sockets Layer(SSL)/Transport Layer Security(TLS) を用いて事前に共通鍵の共有が行われている．DC-NTM 端末間の通信は,AS から配布される共通鍵で暗号化が行われている．このため,Route Direction メッセージ及び Relay Direction メッセージは暗号化されている．既存方式で用いられる 3 種類の暗号鍵の詳細を以下に示す．

- エンド鍵 (End Key)
  トンネル通信の暗号化に用いられる暗号鍵であり,NTM 端末により生成される．エンド鍵は,Tunnel Request メッセージで相手 NTM 端末へ配布される．
- トンネル鍵 (Tunnel Key)
  エンド鍵を配送する Tunnel Request メッセージを暗号化する鍵である．トンネル鍵は DC により生成され,Route Direction メッセージにより NTM 端末に配送される．また,Tunnel Request メッセージを中継する RS にも配布される．
- 一時鍵 (Temporary Key)
  NTM 端末間の通信の暗号化に用いるエンド鍵を交換する際に利用する暗号鍵であり,DC により生成される．一時鍵は DC から NTM 端末のみに Route Direction メッセージにより配送され,RS には配送されない．そのため,RS を経由するエンド鍵交換時にも, エンド鍵を保護することが可能となる．

既存の方式では,DC が攻撃され一時鍵及びトンネル鍵が流出した場合, エンド鍵の流出に繋がるおそれがある．エンド間通信において, 外部装置の要因により秘匿性が失われることは極めて重大な課題である．

3.　提案方式

本研究では,NTMobile へ PKI を導入することにより, セキュリティの強化を目指す．NTM 端末にディジタル証明書を発行し,NTMobile 上で有効な証明書チェーンを形成する．ディジタル証明書の公開鍵を用いて一時鍵の暗号化を行う．上記により, 外部装置に依存しないエンド鍵の共有が可能になる．

また, ディジタル証明書を参照することで相手 NTM 端末の認証が可能になる．エンド間通信では, 相手端末の IP アドレスが変化するなどのため, 相手端末が正規のノードであることを保証することが困難である．ディジタル証明書を用いることで, 自身が通信したい相手であることを確認可能である．

Vol.2017-DPS-171 No.30
Vol.2017-MBL-83 No.30
Vol.2017-ITS-69 No.30
2017/6/2

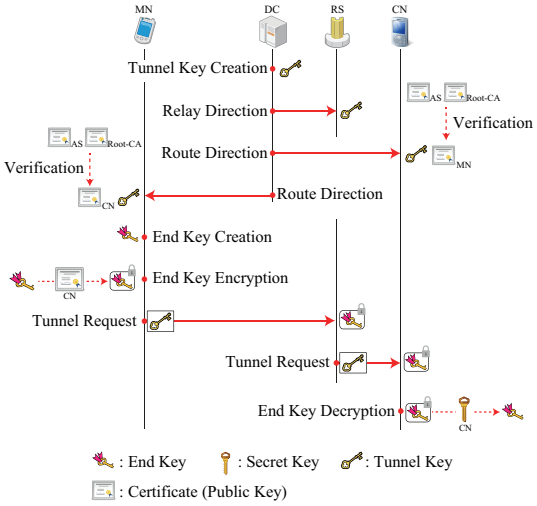

図 3 ディジタル証明書を用いたエンド鍵の共有手法

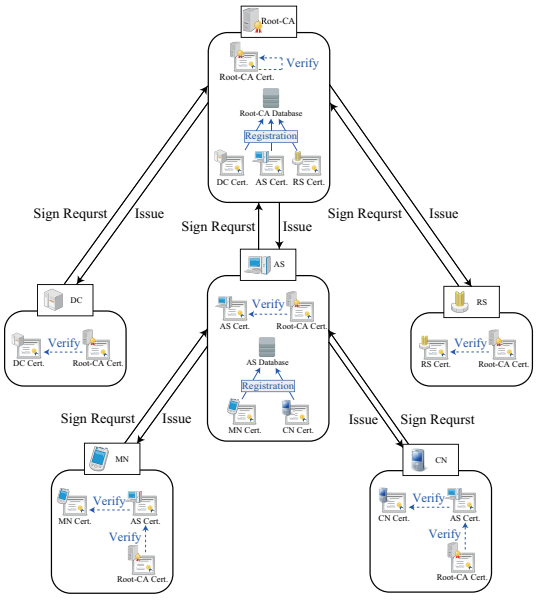

図 4 拡張する証明書チェーン

3.1 PKI の導入

図 3 に, 提案手法におけるエンド鍵の共有方法を示す. 既存方式では,DC から配布される一時鍵を用いてエンド鍵を暗号化していた. 提案では一時鍵を廃止し, ディジタル証明書の公開鍵による暗号化を行う. 暗号化されたエンド鍵を復号できる端末は通信相手の NTM 端末のみである.

図 4 に提案する証明書チェーンの構図を示す. NTM 端末へディジタル証明書を発行するにあたり,NTMobile ネットワークで有効な中間認証局の設置を行う. 提案では,AS に中間認証局の機能を付加するものとする. AS が NTM 端末へディジタル証明書を発行し,PKI の管理を行う. また, ディジタル証明書の利用にあたり Certificate Revocation List(CRL) の導入が必要である. CRL はディジタル証明書の失効リストであり, 認証局が任意で失効した証明書のシリアルナンバーを記録するものである. 提案において CRL と同等の機能を提供する手法をあわせて説明する. 以下に提案における NTMobile の変更点を示す.

- NTMobile における FQDN の仕様変更
  NTMobile の構成装置郡と NTM 端末に割り当てられる FQDN の最大値を 64byte へ変更する. 既存の NTMobile では,FQDN の割り当てを RFC に準拠しており 255byte を上限としている. 一般に用いられる X509v3 証明書では,FQDN を記載するフィールドとして CommonName フィールドが設けられている. しかし, 当該フィールドは最大長が 64byte となっているため,FQDN の割り当てを 64byte 以下へ変更する必要がある. 提案ではドメイン部の最大値を 31byte とし, ホスト部の最大値を 32byte と定義する. また, ホスト部とドメイン部を区切るドットに 1byte 割り当てるものとする.
- AS の変更点
  ディジタル証明書の発行及び失効の機能を付加する. 初回に NTM 端末から AS へ認証が行われるが, この処理でディジタル証明書の有効性を確認する. NTM 端末のディジタル証明書を発行及び失効する際には, 該当 DC へ通知を行う. また,NTM 端末のログイン時にもディジタル証明書の有効性を通知する. 通知では, 発行及び失効したディジタル証明書のシリアルナンバーと有効性を示すステータスを共有する. 通知はリアルタイムで行われ,DC と AS 間で常に最新の証明書ステータスが共有される.
- DC の変更点
  経路構築時にディジタル証明書の有効性を確認できるよう拡張を行う. DC のデータベースを拡張し,NTM 端末の保持するディジタル証明書のシリアルナンバーとステータスを記録する. 経路構築時には,AS から通知されたシリアルナンバーとステータスを参照した上で指示を行う. この拡張により, リアルタイムでディジタル証明書のステータスを確認可能である. 既存の NTMobile のシグナリングを大きく変更することなく, ディジタル証明書を活用可能となる.
- NTM 端末
  NTM 端末の保持するディジタル証明書には, 自身に割り当てられた FQDN を記載する. また,NTM 端末のディジタル証明書の検証にはルート証明書と AS のディジタル証明書が必要になる. これは中間認証局を設置したためである. 中間認証局のディジタル証明書とルート証明書は事前に保持しておくものとする.

3.2 シグナリングの拡張

NTMobile のシグナリングは 2 種類に大別可能である. 起動時に行われるログイン処理と通信時に行われる経路構築処理である. 経路構築処理については, 構成要素との連携を行う経路選択処理と相手 NTM 端末との連携を

Vol.2017-DPS-171 No.30
Vol.2017-MBL-83 No.30
Vol.2017-ITS-69 No.30
2017/6/2

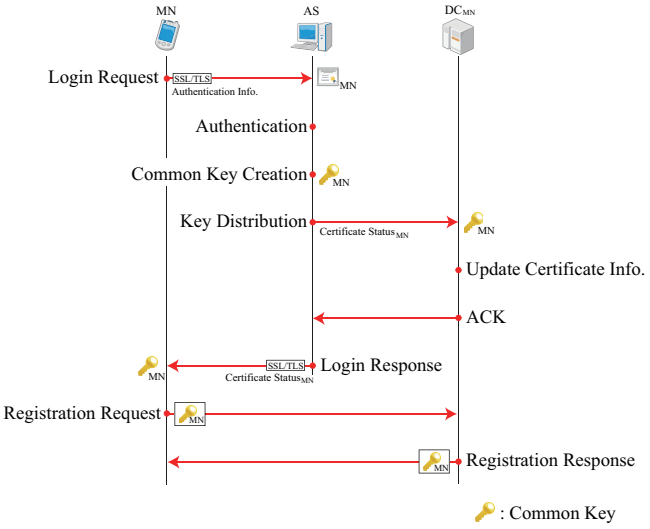

図 5　ログイン処理の概要

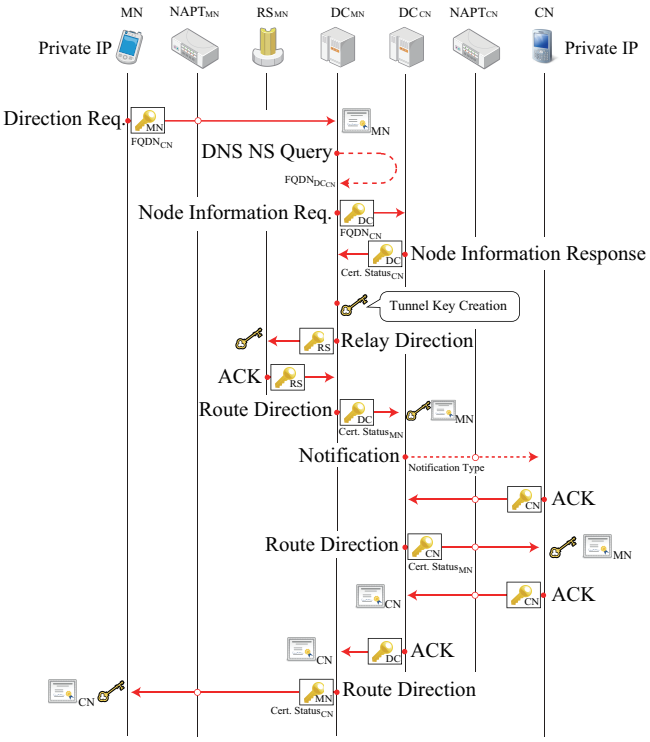

図 6　経路選択処理の概要

行うトンネル構築処理に分けられる．ここで通信開始側 NTM 端末を Mobile Node(MN), 通信相手側 NTM 端末を Correspondent Node(CN) と表記する．また,MN を管理する DC を DC_{MN},CN を管理する DC を DC_{CN} と表記する．シグナリングの拡張点について以下に示す．

- ログイン処理
  図 5 にログイン処理の概要を示す．ログイン処理では,SSL/TLS を用いた通信により AS と NTM 端末間の認証が実施される．認証成功時には DC との連携が開始され, 実 IP アドレスや NAPT のポート番号などの位置情報登録が行われる．既存の NTMobile では,AS-NTM 端末間で片方向認証の SSL/TLS 通信が

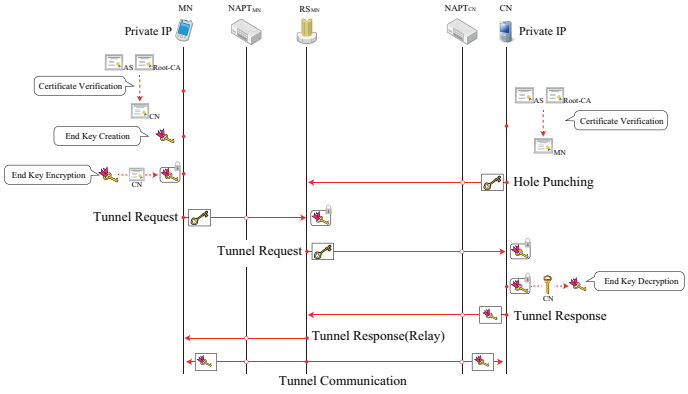

図 7　トンネル構築処理の概要

行われている．提案では, 自身のディジタル証明書を用いることで SSL/TLS の双方向認証が可能となる．AS はディジタル証明書の有効性を確認し, 該当 NTM 端末を管理する DC へ Key Distribution メッセージとして証明書のステータスを通知する．DC は, 得られたステータスを自身のデータベースへ登録する．

- 経路選択処理
  図 6 に経路選択処理の概要を示す．NTMobile の経路構築処理の前半部分にあたる処理である．MN が DC_{MN} へ送信する経路構築の依頼をトリガとして処理が開始される．経路構築依頼は Direction Request として DC_{MN} に送信されるが, ここで自身のディジタル証明書を添付する．DC_{MN} は DC_{CN} と連携を行い,Node Information Request/Response メッセージにより CN の情報を入手する．得られる情報は,CN の位置情報, 仮想 IP アドレス, ディジタル証明書のシリアルナンバーとステータスである．MN と CN の位置情報から適切な経路を決定し,Route Direction メッセージとして MN 及び CN へ通知を行う．Route Direction ではアドレス情報やトンネル鍵を添付するが,DC が記録している MN 及び CN のディジタル証明書の情報を合わせて添付する．また, ディジタル証明書も Route Direction に付加される．CN のディジタル証明書は Route Direction の応答メッセージに添付される．
- トンネル構築処理
  図 7 にトンネル構築処理の概要を示す．経路構築処理の後半にあたるトンネル構築処理では, 相手 NTM 端末と直接通信を行いエンド鍵を共有する処理が行われる．提案では Route Direction によりディジタル証明書を入手しているため, ディジタル証明書の検証を行う．検証は, ディジタル署名, シリアルナンバー,FQDN, 有効期限の 4 点を調べる．Route Direction には,DC が記録している証明書のシリアルナンバーが添付されているため一致するか調査する．検証成功時にエンド鍵を生成し,CN のディジタル証明書の公開鍵で暗号化を行う．暗号化されたエンド鍵を Tunnel Request メッセー

Vol.2017-DPS-171 No.30
Vol.2017-MBL-83 No.30
Vol.2017-ITS-69 No.30
2017/6/2

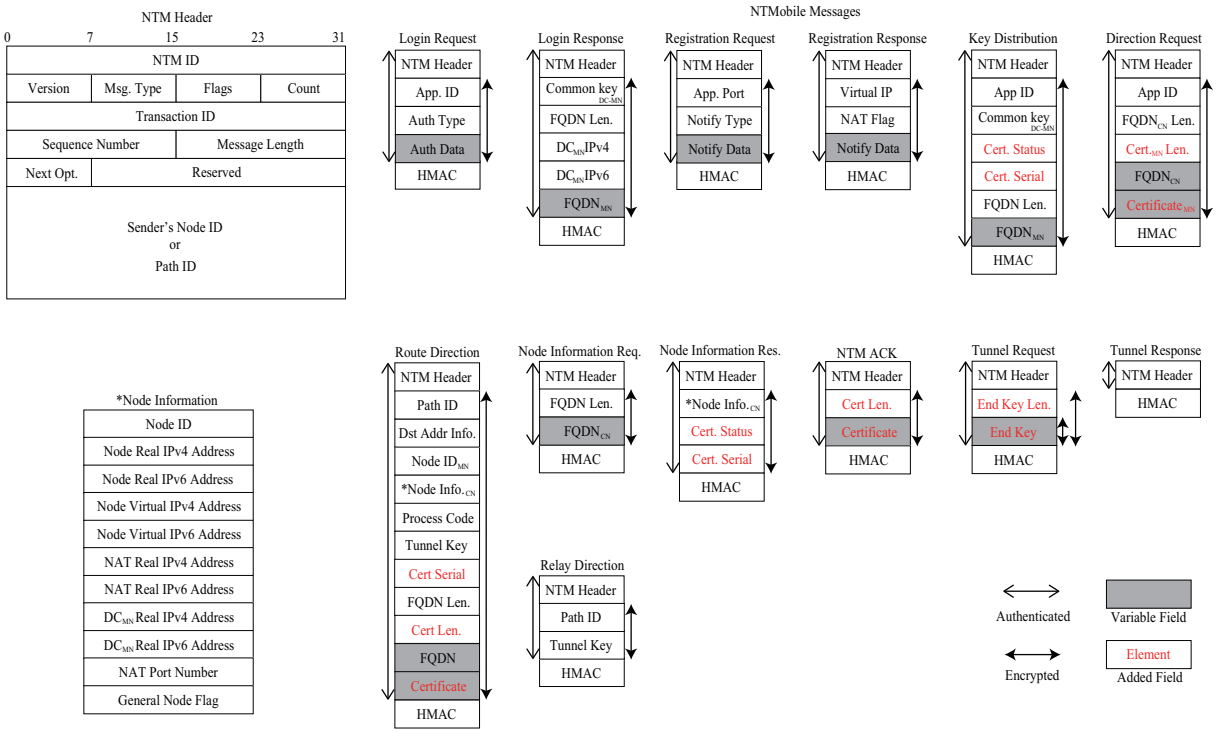

図 8　メッセージフォーマット

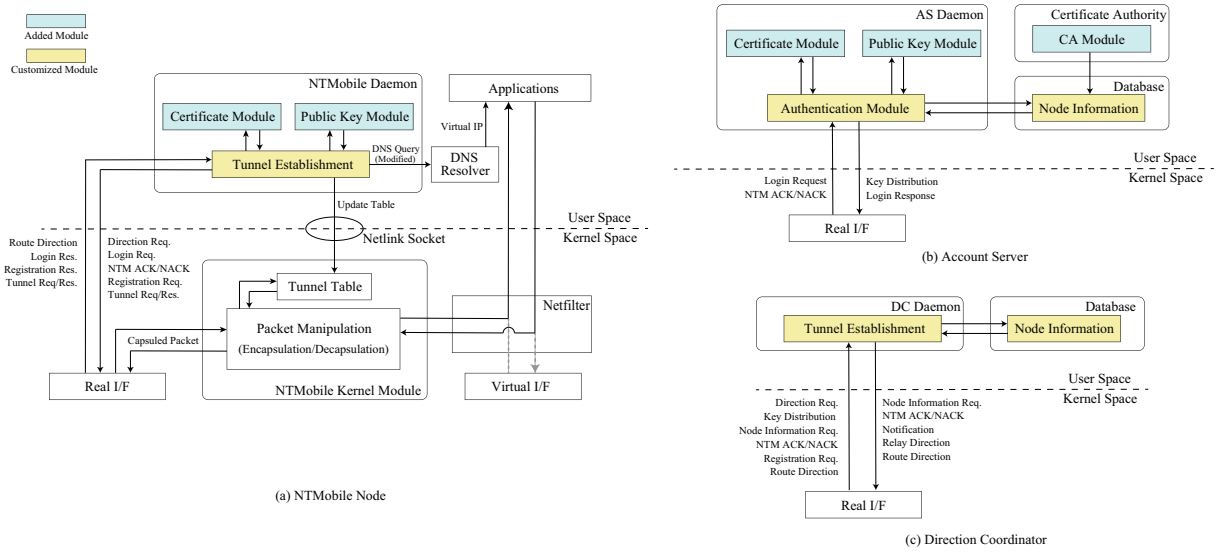

図 9　モジュール構成

ジとして送信し,CN とエンド鍵の共有を行う．CN は受信した Tunnel Request メッセージを自身の秘密鍵を用いて復号しエンド鍵を入手する．

4. 実装

図 8 に定義したメッセージフォーマットを示す．図中の赤字が追加したフィールドである．ディジタル証明書は可変データであるため, 既存のメッセージボディの末尾に付加するものとし, データ長を示すフィールドを追加した．DC-AS 間で共有されるディジタル証明書のシリアルナンバーは,X509v3 形式に準拠し 20 オクテットのフィールドを追加した．ディジタル証明書の有効性を示すステータスフィールドは 1byte で定義した．提案では Tunnel Request メッセージに EndkeyLength フィールドを追加しているが, 既存の NTMobile では当該フィールドは存在しない．共通鍵暗号方式による処理では, 平文と暗号データのサイズが一致するためである．公開鍵暗号方式では, 暗号化対象にパディングを行い暗号化するため, 平文と暗号データではサイズが異なる．このため,Tunnel Request メッセージでは暗号化後のサイズを記録するフィールドを追加した．

図 9 に実装したモジュール構成を示す．NTM 端末及び AS に, ディジタル証明書を操作する Certificate モジュール

Vol.2017-DPS-171 No.30
Vol.2017-MBL-83 No.30
Vol.2017-ITS-69 No.30
2017/6/2

表 1　実験諸元

Host Machine	
OS	OS X 10.9.5
CPU	Intel Core-i7 2.8GHz
Memory	16 GBytes
Software	VirtualBox 5.0
Virtual Machine	
OS	Ubuntu 12.04
Memory	512 MBytes
Certificate	X509v3 SHA256 with RSA
Public key cryptosystem	RSA 2048bit
Common key cryptosystem	AES-CFB 128bit
Number of measurements	10

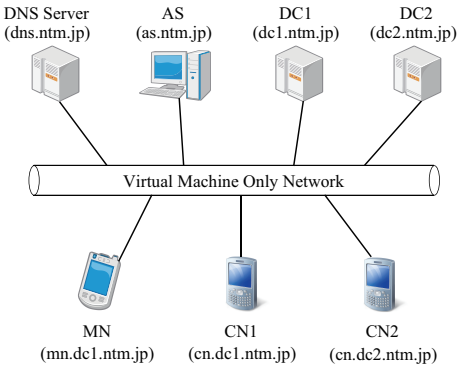

図 10　実験環境

表 2　経路構築完了時間

	Time[ms]	
	Exist model	Proposed model
DC 1 unit	37.8	92.2
DC 2 unit	65.1	138.7

と公開鍵アルゴリズムを使用する PublicKey モジュールを追加した．Certificate モジュールは, ディジタル証明書の検証や形式の変換を行う．PublicKey モジュールは, 公開鍵アルゴリズムを使用するためのモジュールであり, データの暗号化と復号処理を行う．また,DC と AS のデータベースにディジタル証明書の管理フィールドを追加した．記録する内容は, ディジタル証明書のシリアルナンバーとディジタル証明書の有効性を示すフラグである．両者データベースにはノードを管理するテーブルが用意されているため, 新たにフィールドを追加した．AS については中間認証局の機能を保持するため,Certificate Authority(CA) モジュールを用意した．ディジタル証明書の発行及び失効時に,AS のデータベースへ情報を登録するものとする．また,NTM 端末と DC が持つトンネル構築モジュールと AS の認証モジュールでは, メッセージフォーマットの変更に伴う改造を行った．

5. 性能評価

図 10 に構成した仮想マシンネットワークを示す．表 1 に

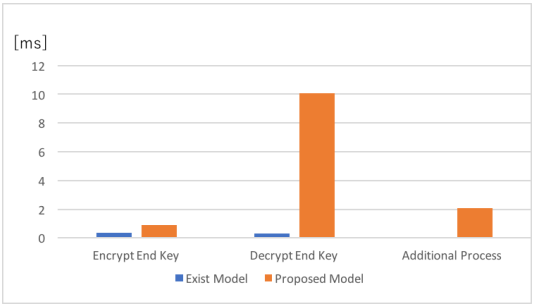

図 11　エンド鍵の暗号復号の処理時間

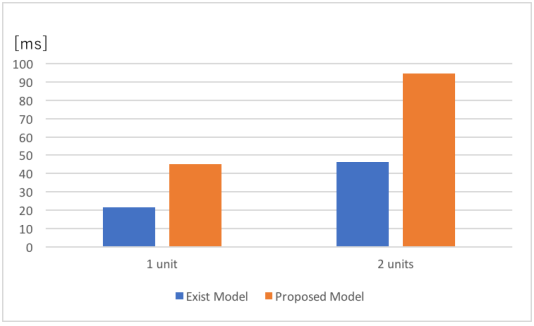

図 12　経路選択処理の処理時間

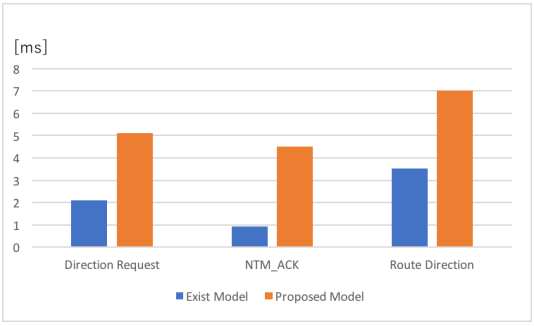

図 13　ディジタル証明書を添付するメッセージの処理時間

作成した仮想マシンとホストマシンの諸元, 使用した測定アプリケーションを示す．米国標準技術研究所 (National Institute of Standards and Technology: NIST) は SSL/TLS の暗号設定に関するガイドラインを発行している [15]．ガイドラインによると, 広く用いられている公開鍵暗号アルゴリズムである Rivest Shamir Adleman(RSA) の鍵長が 2048bit であれば利用可能と定めている．一方で, 代表的な共通鍵暗号アルゴリズムである Advanced Encryption Standard(AES) の鍵長は 128bit 以上であれば利用可能としている．この基準に準拠し,RSA 2048bit,AES 128bit での検証を行う．PKI の機能は, 広く用いられているライブラリである OpenSSL API を用いて実装した．公開鍵暗号アルゴリズムは, 共通鍵暗号アルゴリズムに比べ処理にかかるオーバーヘッドが大きいことが知られている．このため, 測定では暗号化処理及び復号処理に着目し, 既存方式から増大するオーバーヘッドを測定する．また, 通信開始時からトンネルの形成までの時間と増加する処理を測定し, 提案手法の実用性を検証する．NTMobile では,MN と CN が

同一 DC に管理されている場合と他 DC に管理されている場合の 2 種類の通信が想定されるため両者の測定を行う. 測定は 10 回行い平均値を算出し評価を行う.

表 2 に測定結果を示す. また, 図 11,12,13 にそれぞれエンド鍵に対する暗号処理時間, 経路選択処理の測定結果, メッセージ生成の測定結果を示す. 経路の構築完了時間において既存方式と比較し,DC1 台時に約 54.4ms,DC2 台時に約 73.6ms のオーバーヘッドを確認した. まずディジタル証明書を添付するメッセージによるオーバーヘッドは全体のオーバーヘッドの 50%を占めることがわかった. ディジタル証明書が添付されるメッセージの生成から送信に要する時間に約 3ms のオーバーヘッドを確認できる. このオーバーヘッドについて, メッセージの暗号化や Message Authentication Code(MAC) の算出によるものであると考えられる. 既存のメッセージボディが数十 byte であるが, ディジタル証明書のサイズは 1000byte 以上であるため, 上記処理に伴うオーバーヘッドが発生したと言える. 受信時にも MAC の算出及び復号の処理を行うことから, 送受信で合計約 6ms 以上のオーバーヘッドが発生する. ディジタル証明書が添付されるメッセージは,DC1 台時は 4 回,DC2 台時は 6 回やりとりされる. よって,DC1 台時には 24ms 程度,DC2 台時には 36ms 程度のオーバーヘッドが発生する計算となる. 図 12 の測定結果は,Direction Request メッセージを受信してから MN へ Route Direction メッセージを送信するまでの時間である. DC は前述のメッセージの処理によるオーバーヘッドの影響を大きく受けるため, 経路指示における時間増加が発生した形となる.

次にエンド鍵の処理に注目すると, トンネル構築処理で 14ms 程度のオーバーヘッドが発生することがわかった. 公開鍵による暗号化は大きな影響を与えないことがわかるが, 秘密鍵による復号処理で約 10ms のオーバーヘッドを確認できる. また図中の Additional Process は, ディジタル証明書の検証や公開鍵の抽出など提案での追加処理の合計である. MN と CN で追加処理の並列化を行っていないことから, 全体で 2 倍の 4ms のオーバーヘッドが発生する. 両者を合計すると約 14ms となる.

既存方式と比較し約 54ms と約 73ms のオーバーヘッドを確認したが, 実環境の無線通信では数 100ms の遅延時間が発生する. このため, 今回のオーバーヘッドは十分に小さい値であると考えられ,NTMobile へ導入可能であるといえる.

6. まとめ

本稿では,NTMobile へ PKI を導入する手法について設計を行い実装評価を行った. 提案により, エンド端末間の通信安全性が向上する上, 相手端末の認証が可能になることを示した. また, 既存の NTMobile を拡張実装することにより, 提案方式が大きなオーバーヘッドなしに実装可能であることを確認した.

謝辞 本研究の一部は科研費 (15H02697, 17K00142) および公益財団法人内藤科学技術振興財団の助成を受けたものである. 記して謝意を表する.

参考文献

[1] APNIC: IPv4 exhaustion details., , available from ⟨https://www.apnic.net/community/ipv4-exhaustion/ipv4-exhaustion-details/⟩
[2] Levkowetz, H. and Vaarala, S.: Mobile IP Traversal of Network Address Translation (NAT) Devices, Technical report (2003).
[3] Society, I.: IP Addressing Issues, , available from ⟨http://www.internetsociety.org/ip-addressing/⟩
[4] Niazi, S. and Dowling, J.: Usurp: Distributed NAT Traversal for Overlay Networks, *Proceedings of the 11th IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems* (2011).
[5] J. Rosenberg, R. Mahy, D. W.: Session Traversal Utilities for NAT (STUN) Session Traversal Utilities for NAT (STUN) Session Traversal Utilities for NAT (STUN) Session Traversal Utilities for NAT (STUN), *RFC 5389* (2008).
[6] Rosenberg, J.: Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols, *RFC 5245* (2010).
[7] ISHIYAMA, M., KUNISHI, M., UEHARA, K., ESAKI, H. and TERAOKA, F.: LINA: A New Approach to Mobility Support in Wide Area Networks (Special Issue on Internet Technology), *IEICE transactions on communications*, pp. 2076–2086 (2001).
[8] D. Le, X. F. and Hogrere, D.: A review of mobility support paradigms for the internet, *IEEE Communications surveys*, pp. 38–51 (2006).
[9] Soliman, H.: Mobile IPv6 Support for Dual Stack Hosts and Routers, *RFC 5555* (2009).
[10] C. Perkins, D. Johnson, J. A.: Mobility Support in IPv6, *RFC 6275* (2011).
[11] Naito, K., Nishio, T., Mori, K., Kobayashi, H., Kamienoo, K., Suzuki, H. and Watanabe, A.: Proposal of seamless IP mobility schemes: Network traversal with mobility (NTMobile), *Global Communications Conference (GLOBECOM)*, pp. 2572–2577 (2012).
[12] Naito, K., Mori, K., Kobayashi, H., Kamienoo, K., Suzuki, H. and Watanabe, A.: End-to-end IP mobility platform in application layer for iOS and Android OS, *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*, pp. 92–97 (2014).
[13] Verble, J.: The NSA and Edward Snowden: Surveillance in the 21st Century, *ACM SIGCAS Computers and Society*, Vol. 44, No. 3 (2014).
[14] Consortium, I. A.: IoT Security Guidelines Ver. 1.0, , available from ⟨http://www.iotac.jp/wp-content/uploads/2016/01/IoT-Security-Guidelines$_{v}$*er*.1.0.*pdf*⟩
[15] Barker, E. and Roginsky, A.: Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths, *NIST Special Publication*, Vol. 800-131A, pp. http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800–131Ar1.pdf (2015).