アドホックネットワークOLSRにおけるセキュア・ルーティング方式の開発

福澤寧子[†,††] 石田修一[†]
安藤英里子[†] 松本勉[††]

アドホックネットワークは，無線通信手段を有するPDA，携帯電話，ノートPCなどの端末によって，集まったその場で，簡易にネットワークを構築できることから，ユビキタス社会の実現を支える技術の1つとして注目されている．しかし，電波が受信可能な範囲では誰もがデータを受信できる無線通信であることから，セキュリティの確保が不可欠である．そこで，アドホックネットワークの不正利用，データ盗聴・改ざんを防止することを目的に，アドホックネットワークのルーティング方式の1つであるOLSR（Optimized Linking State Routing）において，セキュリティオーバヘッドが少ないセキュア・ルーティング方式を開発した．直接通信可能な端末間で端末間認証，鍵共有を行い，共有した鍵に基づく制御メッセージの完全性保証を行うことで，不正な端末がルーティングテーブルに登録されることを防ぐとともに，セキュア・ルーティングの過程で全端末が同じ鍵を共有する仕組みを有することで，端末の移動にともなう認証処理を20%以下に低減できる方式である．

Development of Secure Routing Mechanism for the Mobile Ad-hoc Network OLSR

Yasuko Fukuzawa,[†,††] Shuichi Ishida,[†] Eriko Ando[†]
and Tsutomu Matsumoto[††]

A Mobile Ad-hoc Network (MANET) is a collection of devices, such as PDAs, mobile phones, portable computers, etc., equipped with wireless communications and networking capability. Such mobile nodes can communicate with other nodes, either within their radio range by direct transmission, or outside their radio range, using intermediate nodes for packet relaying process. In this paper we propose security mechanism for the Optimized Linking State Routing, which is one example of a proactive routing protocol for MANET. Our main approach is based on mutual authentication with direct communicate nodes to connect network, and checking of the integrity of control message for securing the network. In particular, all nodes can share a shared key in the process of secure routing control. This mechanism reduces by more than 20% the authentication process accompanying terminal movement.

1. はじめに

携帯電話や無線LANなど，無線通信技術が社会に広く浸透してきている．その中で，固定的な通信基盤施設を必要とせず，モバイル端末（以下，端末）が協調して動的にマルチホップの無線ネットワークを構築するモバイルアドホックネットワーク（以下，アドホックネットワーク）は，いつでもどこでもネットワークを利用可能とするユビキタス社会の実現では，欠かせない技術の1つである．固定的な通信基盤施設を設置せずに，端末を持ち寄るだけでネットワークを構築できるため，ITS（Intelligent Transportation system：高度道路交通システム）や映像監視システムなどにおける一時的なネットワーク構築に期待されている．

ところで，アドホックネットワークでは，端末が互いに無線通信領域内にありデータパケットが直接送受信される場合だけでなく，電波が直接届かない場所にいる端末どうしでもデータ送受信を可能にするために，各端末は他の端末が送信するデータパケットを中継する転送機能を有する必要がある．IETFのMANET（Mobile Ad-hoc Networks）[1]では，通信経路を動的に決定するためのアドホックネットワークのルーティングプロトコルとして複数の標準化案[2)~6)]が検討されている．

† 株式会社日立製作所システム開発研究所
Systems Development Laboratory, Hitachi, Ltd.
†† 横浜国立大学
Yokohama National University

通信を行う際に必要な経路を探索するリアクティブ型プロトコルでは，送信元端末はルートリクエストをブロードキャストし，受信した端末は，さらに，それをブロードキャストすることでルートリクエストを伝播させ，送信先端末はルートリプライを返送する．このため，通信経路を得るためにネットワークの帯域を浪費しないが，ルート探索に時間を要するため，データ送信に遅延が生じる．MANET においては DYMO（Dynamic MANET On Demand Routing）[6] が標準化の有力候補になっている．

プロアクティブ型プロトコルでは，定期的に接続情報を交換することで，送信先への経路情報を常時保持する．このため，データ送信を行う際にすぐに経路が分かるが，経路情報の交換にネットワークの帯域を消費する．MANET においては OLSR [2] が標準化の有力候補になっている．

一方，無線ネットワークでは電波到達範囲に位置すれば誰もがデータを受信できることから，特に情報の取扱いが厳しいシステムの実用化には，アドホックネットワークの不正利用を防止するセキュリティ対策が不可欠である．無線通信として無線 LAN（IEEE802.11）などが用いられるが，従来の無線 LAN のセキュリティ仕様[7] で安全性が確保できるのは，認証サーバを使用したアクセスポイントと端末間，あるいは事前に鍵を共有したアクセスポイントと端末間である．このため，OLSR をはじめとするアドホックネットワークでは，サーバが不在であること，また，接続する端末が流動的であることから接続する可能性のある全端末との間で事前に鍵を共有するセキュリティ仕様は適さない．

そこで，OLSR において，不正な端末がアドホックネットワークに接続，不正利用することを防ぐことを目的としたセキュア・ルーティング方式としては，相互認証などは行わず，OLSR の制御メッセージのパケットに非対称鍵暗号による署名と公開鍵証明書を付加することで制御メッセージの改ざんを防ぐ方式（以下，署名付加方式）[8] が提案されている．署名付加方式は，たとえば，ITS における交通情報の取得を目的に，高速で走行する車両がアドホックネットワークに接続し，瞬時に情報を取得し，接続を開放するようなアドホックネットワークの利用においては，アドホックネットワーク構築の際に相互認証処理を行わない署名付加方式は有効である．

しかし，アドホックネットワークでは，動的ルーティングをするために，各端末が頻繁に（たとえば 2 秒に 1 回程度）制御メッセージを送出するので，すべての制御メッセージのパケットに非対称鍵暗号系の署名を付与する署名付加方式は，セキュリティ処理にともなう端末の負荷が高い．たとえば，交通情報や大量の映像情報などの情報取得において，継続的なアドホックネットワーク接続を要するやや存命時間が長いアドホックネットワークの一般的な利用においては，アドホックネットワークへの参加確立後（以下，定常時と称する）もセキュリティオーバヘッドが高いものとなる．したがって，署名付加方式は，アドホックネットワーク OLSR の一般的な利用に十分応えられるものにはなっていない．

そこで，本論文では，アドホックネットワーク OLSR 向けに，定常時の負荷が小さく，低リソースの端末やネットワーク構成の激しい変化に対応する軽量なセキュア・ルーティング方式を提案する[9),10)]．

以下，2 章ではアドホックネットワーク OLSR のセキュリティ要件，3 章では OLSR セキュア・ルーティング方式の提案，4 章では OLSR セキュア・ルーティング機能の開発および評価を述べる．

2. アドホックネットワーク OLSR のセキュリティ要件

OLSR におけるセキュア・ルーティングの基本設計要件を明らかにするために，OLSR の概要を述べたうえで，OLSR に想定される脅威を洗い出し，セキュリティ要件，実現上の要件を示す．

2.1 OLSR の概要

図 1 は OLSR の動作を概念的に示した図であり，IP アドレスによって端末を識別，制御する．OLSR は，データ通信の有無にかかわらず，定期的に制御メッセージ（HELLO，TC：Topology Control）を交換し，あらかじめ経路表を作成しておくプロトコルである．各端末は，直接通信可能な端末の情報を HELLO として，定期的（たとえば 2 秒ごと）に配信する．HELLO を受信した端末は，HELLO 送信元端末を直接通信可能な端末と把握し，また，送信元端末を介して間接的に通

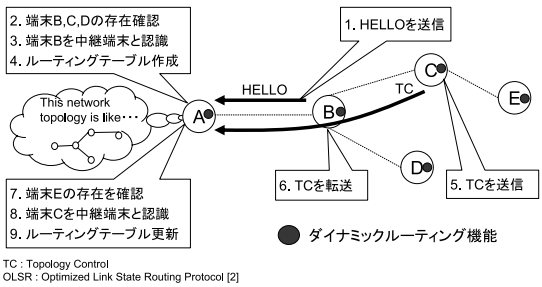

図 1 OLSR 方式の概要
Fig. 1 Outline of OLSR.

表 1 脅威分析
Table 1 Threat analysis.

#	攻撃	手段	脅威・影響	対策
1	盗聴	正当端末の側に近づき、APデータを受信	不正端末に情報が漏洩	APデータの暗号化（オプション）
				セキュリティデータの暗号化
2	偽造、改ざん	不正端末が偽りのAPデータを送信	正当端末が正しい情報を取得不可	APデータ受信時の改ざん検知（オプション）
		不正端末が偽りの制御データを送信	不正なルーティングテーブルが生成され、通信不可（不正なアドホックネットワークが構築）	制御メッセージ受信時の改ざん検知と転送およびルーティングテーブルへの登録制御
3	なりすまし	不正端末が正当端末になりすましてネットワークに参加、不正端末間でAPデータ送受信	不正端末が正当端末の帯域を利用し、正当端末の利用可能な帯域が減（アドホックネットワークの不正利用）	認証による制御メッセージの受信制御
		不正端末が正当端末になりすまして、APデータをブロードキャスト送信		APデータ受信時の受信制御（オプション）
		不正端末が正当端末になりすまして、APデータ送信先の不正端末の側の正当端末宛に送信		

APデータ：アプリケーションデータ

信できる端末を把握する．受信端末は，受信端末が以降に発する HELLO において，間接的に通信するために中継してほしい端末を中継端末 MPR（Multi Point Relay）として指定する．HELLO によって MPR に指定された各端末は，自端末を MPR と指定した端末の情報をまとめて TC として定期的（たとえば 5 秒ごと）に配信し，TC を受信した各 MPR はこれを転送する．このように，すべての端末は HELLO と TC によりネットワークトポロジを把握し，各端末それぞれがルーティングテーブルを作成，更新する．

2.2 セキュリティ要件

OLSR の処理に基づき想定される脅威を洗い出し，セキュリティ要件を明らかにする．

2.2.1 脅威洗い出しの前提

制御データと通信データを資産とし，第 3 者による脅威を洗い出す．端末装置内のメモリなどの LSI を直接解析する物理的攻撃，DoS（Denial of Service Attack）攻撃や無線妨害などの物理的な攻撃は本検討の範囲外とする．ここでは，各端末に搭載されたプロトコルが正しく動作するものとし，セキュリティ情報は耐タンパ性が保証された SAM（Security Application Module）によって管理するものとする．

2.2.2 脅威と対策

表 1 に脅威と対策を洗い出した結果を示す．

(1) 盗聴：無線通信では，電波到達範囲内の端末は通信路上のデータを受信し，盗聴できるため，通信路上のデータが第 3 者に漏洩する．このため，暗号化を行う必要があるが，ここではセキュリティ情報のみを秘匿対象とする．

(2) データの偽造，改ざん：電波到達範囲内の端末に対して，たとえば，不正端末が偽りの制御メッセージを送信した場合，正当な端末が不正な制御情報から不正なルーティングテーブルを作成し，不正なネットワークを構築してしまい，正当な端末間での通信ができなくなる．そこで，制御メッセージの改ざん検知機能により，改ざんのないことを確認したうえで制御メッセージの転送やルーティングテーブルへの登録を行う．

(3) なりすまし：直接通信できない端末間でデータを送受信するために，中継端末がデータを転送する．このため，たとえばアドホックネットワークに隣接する不正端末 A が正当な端末になりすまし，数ホップ先の正当端末 C 宛に不正端末 B 宛の内容のメッセージを送信すると，正当端末 C の近くに存在する不正端末 B はメッセージを傍受できる．不正端末間でネットワークを不正に利用することにより，正当な端末に提供されるべき帯域が狭くなる．そこで認証により通信相手を確認し，認証済み端末の制御メッセージのみ受信する．

2.2.3 実現上の要件

前述のセキュリティ対策を実現するうえでの課題は以下である．

(1) 認証：アドホックネットワークでは接続関係が流動的であり，接続する可能性のあるすべての端末間で認証のための鍵を事前に共有することは困難である．また，認証サーバなどの固定的ノードは仮定できない．

(2) 制御メッセージの改ざん検知：アドホックネットワークでは定常状態でも制御メッセージを頻繁に送出するので，制御メッセージの改ざん検知にともなうセキュリティ処理の負荷を小さくする必要がある．

(3) 移動時認証：アドホックネットワークでは，端末の移動にともない，接続関係に変更が生じ，認証済み端末からなるネットワークのトポロジが変化する．ネットワークトポロジの変化にともなうセキュリティ負荷は小さくする必要がある．

3. OLSR セキュア・ルーティング方式の提案

2 章に記載したセキュリティ要件を満たす OLSR を，OLSR セキュア・ルーティング方式と称し，実現上の要件を満たす方式を提案する．また，提案方式の有効性を示す．

3.1 提案方式の概要

(1) 認証：アドホックネットワークに参加する際に，直接通信可能な隣接端末間で，非対称鍵暗号による相互認証を行い，端末の正当性を確認する．非対称鍵暗号系の鍵は，端末間で事前に共有する必要がない．各端末は固有の非対称暗号鍵と，公開鍵証明書検証用の鍵を，あらかじめ有しているものとする．

(2) 制御メッセージの改ざん検知：制御メッセージには，非対称鍵暗号系の署名ではなく，処理が軽量な対称鍵暗号系の MAC（Message Authentication Code）を付加する．制御メッセージは直接通信可能な端末にブロードキャストされるので，各端末は隣接端末間との認証時に MAC 生成およびアドホック鍵秘匿用の鍵（以下，リアル鍵）を配布する．各端末は，受信した制御メッセージの完全性を検証したうえで，ルーティングテーブルの生成や制御メッセージの転送を行う．

(3) 移動時認証：認証された端末からなるアドホックネットワークを構築する際に，全端末で共通の対称鍵暗号系の鍵（以下，アドホック鍵）を共有し，端末が移動した場合はアドホック鍵で認証する．アドホックネットワークでは鍵配布サーバを仮定できないので，各 MPR がアドホック鍵を生成し，リアル鍵でアドホック鍵を秘匿して，鍵識別子とともに TC で配布する．各端末は，複数の MPR が生成するアドホック鍵を受信することになるので，生成時刻などの選択基準で鍵を選択保管し，鍵識別子とともに利用する．これにより，各端末は TC によってネットワークトポロジを把握すると同時に，アドホック鍵を共有する．以降，各 MPR が生成する TC には，保管したアドホック鍵を搭載することで，最終的には 1 つの鍵を共有できる．

異なるアドホック鍵を持つ他のネットワークとの間で新規の接続が生じた場合に，TC によってアドホック鍵が統一されるまでの間も，接続前のネットワーク関係を維持するには接続前のアドホック鍵を保持する必要がある．保持していない場合には，移動時にも非対称による認証を行うことになる．

表 2 方式比較
Table 2 Comparison of security countermeasure for OLSR.

攻撃	提案方式	署名付与方式
制御メッセージ盗聴	暗号化による配布鍵情報の秘匿	ー
制御メッセージ改ざん	制御メッセージへのMAC付与とルーティングテーブルの登録制御	制御メッセージへの署名付与とルーティングテーブルの登録制御
端末のなりすまし	相互認証とルーティングテーブルの登録制御	制御メッセージへの署名付与とルーティングテーブルの登録制御

3.2 提案方式の有効性の評価

提案方式と従来方式である署名付加方式を，データ量，安全性，処理性能の観点で比較することにより，提案方式の有効性を明らかにする．

3.2.1 評価の前提

評価の前提を示す．

(1) 署名付加方式の概要

署名付加方式では，すべての制御メッセージ（HELLO，TC）に対して非対称鍵暗号を利用した認証コード（署名）を付与する．署名には，タイムスタンプ情報を付加し，リプレイアタックを防止する．また，公開鍵証明書をあわせて配布する．署名により制御メッセージの改ざん検知と送信元の確認を行っており，署名付加方式はセキュリティ要件を満たす．

提案方式と署名付加方式の比較を**表 2** に示す．

(2) 暗号アルゴリズム

非対称鍵暗号系，および対称鍵暗号系のアルゴリズムと性能は以下とする．なお，性能は実測値である．

- プラットフォーム：
  CPU: Intel(R) ☆1 PXA2（400 MHz）
  OS: Embedded Linux ☆2（Open PDA ☆3）
  暗号ソフト：OpenSSL 0.9.7d [11)]
- 非対称鍵暗号系（RSA ☆4 1,024 ビット）
  署名 R-Sign/復号：106 msec
  検証 R-Ver/暗号化：6 msec
- 対称鍵暗号系（AES 128 ビット）
  署名/検証 A-Sign/A-Ver：4 msec
- MAC（C-MAC AES）：4 msec
- SHA-1：34 msec
- 相互認証 R-Auth：(R-Sign + R-Ver × 2) × 2

☆1 Intel は，米国およびその他の国における Intel Corporation またはその子会社の商標または登録商標です．
☆2 Linux は，Linus Torvalds の米国およびその他の国における登録商標あるいは商標です．
☆3 Open PDA は，米国 Metrowerks Corporation の登録商標または商標です．
☆4 RSA は，RSA Security Inc. の登録商標です．

表 3 セキュリティ機能実現のためのデータ
Table 3 Data for security functions.

	各ノードの保有データ(Byte)		各ノードの秘密管理データ(Byte)		パケットデータ(Byte)	
署名付加方式	公開鍵証明書	1000	秘密鍵	128	Hello	128+1000
	CA 証明書	1000			TC	128+1000
提案方式	公開鍵証明書	1000	秘密鍵	128	Hello	12
	CA 証明書	1000	ｱﾄﾞﾎｯｸ鍵×保有数	16×m	TC	12+48
			リアル鍵×2世代×直接通信ﾉｰﾄﾞ数	16×2×n	相互認証(非対称鍵暗号)	2578
					相互認証(対称鍵暗号)	352

- 相互認証 A-Auth：(A-Sign + A-Ver) × 2

3.2.2 データ量

表 3 は，それぞれの方式を実現するうえで必要となるデータ量を示す．

(1) 各端末があらかじめ保有しておくデータ量

両方式ともに，各端末は端末固有の非対称鍵暗号系秘密鍵に対応する公開鍵証明書，および，証明書を検証する CA 証明書をあらかじめ保有する．証明書は 1,000 B とする．

(2) 各端末が秘密に管理するデータ量

両方式ともに，各端末は端末固有の非対称鍵暗号系秘密鍵を保有する．これに加え，提案方式では，各端末は，リアル鍵は新旧 2 世代 32 B，リアル鍵は直接通信端末数分保有する．アドホック鍵は 16 B × m を保有する．アドホック鍵は複数 m 保有することが望ましい．

(3) パケットで送受されるデータ量

署名付加方式では，全制御パケットに署名と証明書を付与するため，パケットあたり 1,128 B のデータを付加する．一方，提案方式では，HELLO には MAC 生成鍵の識別子 4 B と MAC 8 B が付与され，TC には MAC 生成鍵の識別子 4 B と MAC 8 B とアドホック鍵 16 B × 3 が付加される．また，提案方式では相互認証パケット分が追加になり，4 章に記載の認証プロトコルに基づき算出したところ，非対称鍵暗号による相互認証で 2,578 B，対称鍵暗号による相互認証で 352 B が必要である．認証処理は，直接接続ノード数分実行される．

3.2.3 安全性

両方式の安全性は，暗号アルゴリズムの安全性が保証されているとすれば，秘密情報の管理に依存する．2.1 節で記したように，SAM 前提としていることから，秘密情報と秘密情報を用いた処理は SAM において管理されるため，両方式の安全は同等である．ただし，署名付加方式が非対称鍵暗号系の秘密鍵だけを SAM で管理すればよいのに対し，提案方式は秘密鍵に加え，リアル鍵，アドホック鍵を管理する必要がある．

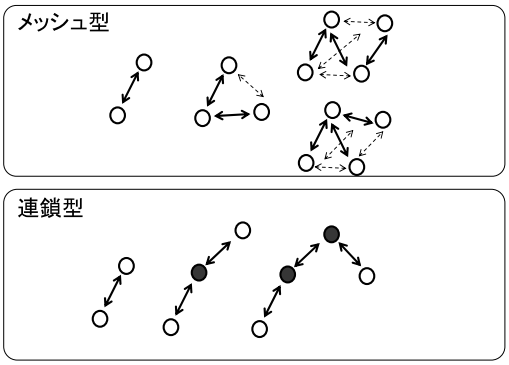

図 2 ネットワークトポロジ
Fig. 2 Network topology.

3.2.4 処理性能

セキュリティ機能実装にともなう負荷を評価するために，(1) 接続時のセキュリティオーバヘッドと，(2) 定常時のセキュリティオーバヘッドを算出し，その端末数依存性を評価する．(1) は，端末がアドホックネットワークに接続する際の平均認証処理時間であり，これはデータ通信確立時間に影響する．(2) は定常時の制御メッセージ改ざん検知処理にともなう端末ごとの処理時間であり，データ通信時の負荷としてスループットに影響する．

図 2 は性能評価のためのアドホックネットワークにおけるネットワークトポロジの基本形である．連鎖型は，直接通信する端末が 2 台であり，重複しない接続でネットワークを構成する．したがって，n 台の端末から構成されるアドホックネットワークでは $n-1$ の直接通信路が存在する．一方，n 台の端末で構成されるメッシュ型では，各端末が $n-1$ 台の端末との間で直接通信路が存在する．メッシュ型は MPR を含まないが，アドホックネットワークで構成しうる密集状態であり，実際のシステムでは，メッシュ型と連鎖型の複合形になる．

以下では，連鎖型とメッシュ型構成のネットワークを評価対象とする．また，全端末が 2 秒間隔で HELLO を送出，MPR は 5 秒間隔で TC を送出する．MPR は受信した TC を転送する．

(1) 接続時のセキュリティオーバヘッド

提案方式は，端末がアドホックネットワークに接続する際に直接通信する端末との間で相互認証を行う．署名付加方式では，接続時に認証などの処理は行わないので，接続時のセキュリティオーバヘッドはない．

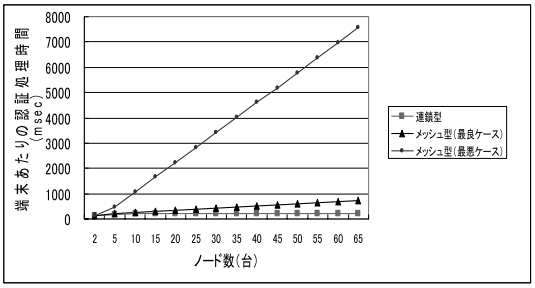

図 3 認証処理時間
Fig. 3 Cost of authentication process.

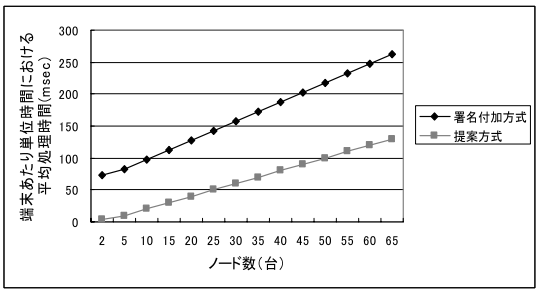

図 4 定常時の端末あたりの制御パケット改ざん検知処理（メッシュ型）
Fig. 4 Cost of integrity confirmation (Type of mesh).

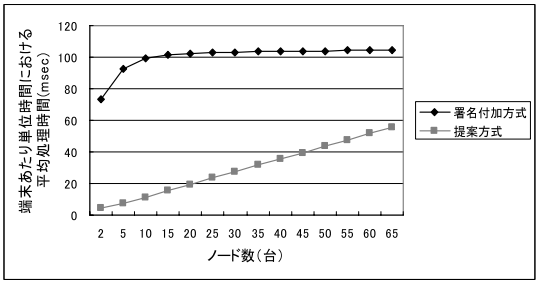

図 5 定常時の端末あたりの制御パケット改ざん検知処理（連鎖型）
Fig. 5 Cost of integrity confirmation (Type of chain).

(a) 連鎖型：n 台の端末から構成される連鎖型のアドホックネットワークでは $n-1$ の直接通信路が存在し，$n-1$ の RSA 相互認証が発生する．したがって，端末あたりの平均認証処理時間は以下となる．

【提案方式】

$$\text{R-Auth} \times (n-1)/n \tag{1}$$

(b) メッシュ型：n 台の端末から構成されるメッシュ型のアドホックネットワークでは $n(n-1)/2$ の直接通信路が存在し，$n(n-1)/2$ の認証が発生し，最良の場合には，$(n-1)$ 回の RSA 相互認証と，$(n(n-1)/2)-(n-1)$ 回の AES 相互認証が生じる．また，最悪の場合には，$n(n-1)/2$ 回の RSA 相互認証が発生する．したがって，端末あたりの平均認証処理時間は以下となる．

【提案方式：最良の場合】

$$\big((n-1)\times\text{R-Auth}+(n(n-1)/2-(n-1))\times\text{A-Auth}\big)\big/n \tag{2}$$

【提案方式：最悪の場合】

$$n(n-1)/2\times\text{R-Auth}/n \tag{3}$$

図 3 は，提案方式の連鎖型とメッシュ型構成における，端末あたりの平均認証処理時間の端末数依存性を示す．

(2) 定常時のセキュリティオーバヘッド

アドホックネットワーク確立後の定常状態における，端末あたり，単位時間あたりの制御メッセージ改ざん検知処理を評価する．

(a) 連鎖型：n 台構成では，n 台の端末が HELLO を送出し，$n-2$ 台の MPR が存在することになる．

【提案方式】

$$\big((\text{MAC}/2\times n)+(\text{MAC}/2\times 2\times(n-1))+\text{MAC}/5\times(n-2)+\text{MAC}/5\times(n-1)\times n+\text{MAC}/5\times(n-3)(n-2)\big)\big/n \tag{4}$$

【署名付加方式】

$$\big((\text{R-Sign}/2\times n)+(\text{R-Ver}/2\times 2\times(n-1))+(\text{R-Sign}/5\times(n-2)+\text{R-Ver}/5\times(n-1)\times n)\big)\big/n \tag{5}$$

(b) メッシュ型：n 台構成のメッシュ型アドホックネットワークでは，TC は発生しない．

【提案方式】

$$\text{MAC}/2+\text{MAC}/2\times(n-1) \tag{6}$$

【署名付加方式】

$$\text{R-Sign}/2+\text{R-Ver}/2\times(n-1) \tag{7}$$

図 4 は，メッシュ型構成における制御メッセージ改ざん検知にともなう端末あたりの平均処理時間を示し，**図 5** は連鎖型構成における同処理時間である．

3.2.5 評価結果

署名付加方式では通信パケットに付加されるデータ量が大きく，提案方式では秘密管理するデータ量が大きくなる．

安全性は，SAM を前提として提案方式と署名付加方式は同等であるが，提案方式では SAM で管理するデータ量が大きいことから，コストが高くなる．

また，提案方式では，図 3 に示す接続時のセキュリティオーバヘッドを必要とする．最良の場合には HELLO の発信間隔（2 秒）以内で相互認証処理を完了できており，データ通信確立までの影響は大きくない．最悪の場合とは，全端末間で同時に認証処理を開始する場合であり，50 端末で端末あたり 6 秒程度の

認証処理時間を必要とする．ただし，同時に認証処理状態になる端末数（たとえば 1 秒以内で認証処理できる端末数）を制限するなどの処理を搭載することで，最悪の場合は容易に回避できる．

一方，図 4，図 5 が示すように，提案方式は定常時のセキュリティ負荷が署名付加方式と比して十分に小さく，スループットを確保するうえで有効であることが分かる．定常時のセキュリティにともなう CPU 負荷を 10%以下にすることを目標にすると，提案方式は，図 4 より 50 台程度のネットワークまで適用可能である．

以上より，データ量/安全性については通信量/格納量で双方に一長一短があるが，アドホックネットワークで重要な性能面では，提案方式は立ち上げ時には認証にともなう負荷があるが，定常時におけるセキュリティ負荷は十分に小さい．したがって，アドホックネットワークの一般的利用には，署名付加方式より，提案方式の方が有効である．

4. OLSR セキュア・ルーティング機能の開発と評価

4.1 開　　発

3 章で述べたセキュア・ルーティング機能を開発した．**図 6** は機能構成であり，動作環境は以下である．

- CPU: Intel(R) PXA2（400 MHz）
- OS: Embedded Linux/Open PDA
- 無線 I/F：802.11b
- 通信プロトコル：TCP/IP
- 暗号ソフト：OpenSSL 0.9.7d [11)]

H-OLSR は（株）日立製作所が開発した OLSR ソフトウェアである．非対称鍵暗号系としては RSA-PSS/OAEP 1,024 ビット，対称鍵暗号系としては AES 128 ビット，SHA-1 と用いた．

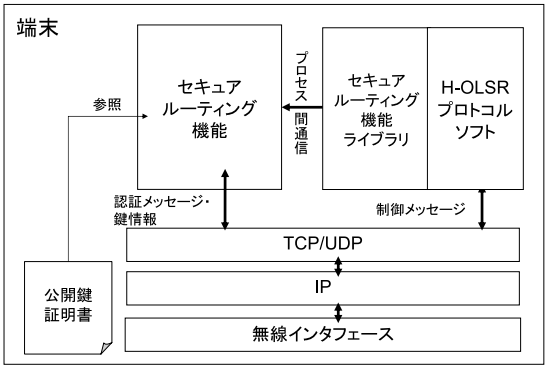

図 6 機能構成
Fig. 6 Structure of function on terminal.

(1) 認証・移動時認証と認証時の鍵共有

認証および移動時認証処理を**図 7** に示す．端末 A は未認証である端末 B から HELLO を受信すると，H-OLSR がセキュア・ルーティング機能を呼び出し，相互認証を開始する．端末 A と端末 B が同じアドホック鍵を共有している場合にはアドホック鍵を，共有していない場合はそれぞれの非対称鍵暗号系の固有鍵による認証を行う．認証は，乱数を用いたチャレンジ・レスポンス方式で行う．

端末 A および B は，それぞれに認証処理中の端末，およびすでに認証に失敗した端末からの制御メッセージ（HELLO，TC）を受信すると破棄する．

端末 A および B は，認証に成功すると双方のリアル鍵およびアドホック鍵を交換する．

なお，リアル鍵を更新の際には，新しいリアル鍵を全隣接端末へ配布完了するまで，更新前のリアル鍵を用いる．そこで，新規参入端末には，新旧のリアル鍵を配布することで，新規参加と同時に MAC 検証が可能になる．また，MPR が存在しないネットワークトポロジの場合には，TC が送信されないので，アドホック鍵が共有できない．そこで，認証時に各端末が保持しているアドホック鍵を交換する．

(2) 制御メッセージの完全性検証

図 8 に示すように OLSR では，複数の制御メッセージから 1 つのパケットを構成することができる．そこで，パケット送信元端末のリアル鍵で，制御パケットの MAC を生成し，付与する．認証に成功した端末からの制御パケットを受信すると，送信元端末のリアル鍵を使って MAC を検証し，制御メッセージの完全性を検証したうえで，制御メッセージにともなう処理を行う．

(3) アドホック鍵共有

各 MPR は，アドホック鍵と，鍵生成時刻や鍵生成端末の IP アドレスを含む識別子を自端末のリアル鍵

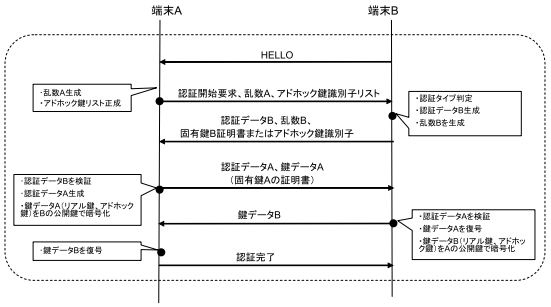

図 7 初期認証・鍵交換処理
Fig. 7 Initial authentication and key distributing.

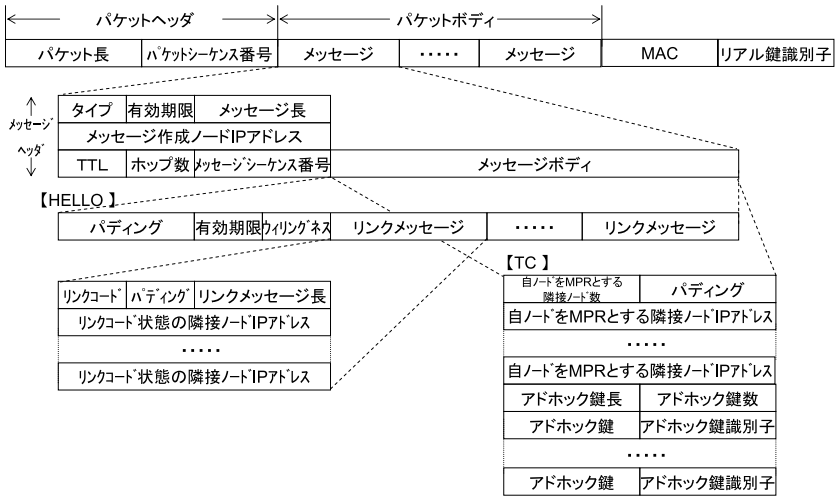

図 8 拡張 OLSR パケットフォーマット
Fig. 8 Advanced format of OLSR packet.

で暗号化し，TC に付与して送信する．

TC を受信すると，送信元端末のリアル鍵で復号し，アドホック鍵などを取得し，受信した TC を自端末のリアル鍵で暗号化し，転送する．TC 送受信のタイミングによって，全端末で同じアドホック鍵を選択するまでにタイムラグが生じる．そのため，最新のアドホック鍵以外の鍵も一定時間または一定数保持することがのぞましい．

(4) 鍵更新

同じ鍵を長期間使用することで鍵情報が漏洩するリスクを下げるため，リアル鍵およびアドホック鍵は，新しい鍵を更新前の鍵で暗号化し，配布する．一定時間が経過すると，新しいリアル鍵を生成し，その時点で認証に成功している隣接端末および認証中の端末に送信する．ここでは，OLSR のシミュレーション[12]に基づくパケットロス率 10.8%を考慮し，3 回の送付により $(1-0.1^3)\times 100 = 99.9\%$のパケット到達が期待できることから一定間隔で鍵更新を 3 回送信する．なお，更新が完了せず送受信者間の鍵の不整合になった場合には，メッセージの完全性保証が不可により，ルーティングテーブルから除き，あらためて端末を検知した段階で初期認証を行う．

アドホック鍵は，更新のタイミングで，MPR が新しいアドホック鍵を生成し，TC で配布する．

4.2 性能評価

開発結果に基づき，OLSR のセキュリティ機能搭載にともなう性能を評価した．

(1) 認証時間

3.2.4 項で示したように，連鎖型よりメッシュ型の方が認証処理にともなう負荷が大きい．そこで，ここではメッシュ型において，1 台の端末に 1〜9 台の端末が存在するネットワーク構成において，非対称鍵暗号系の RSA を用いた初期認証，対称鍵暗号系 AES のアドホック鍵を用いた移動時認証に要する時間をそれぞれ測定した結果を**図 9** に示す．

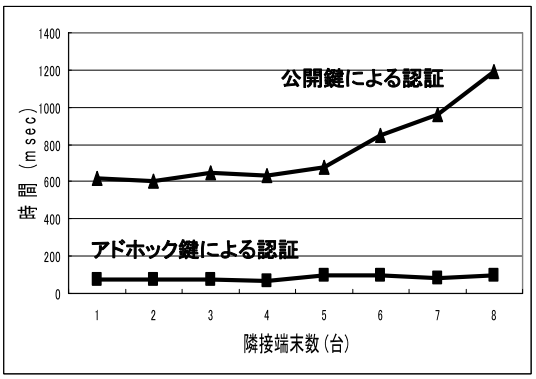

図 9 平均認証処理時間
Fig. 9 Average cost of initial authentication and key distributing.

対称鍵暗号であるアドホック鍵を用いた認証により，初期認証に対する処理時間を 20%以下に低減できた．トポロジ変化が激しいアドホックネットワークにおいてセキュリティ処理を効果的に削減できる．

(2) スループット

表 4 は，端末 10 台が直線的に接続した場合の終端端末間におけるスループットを Netperf[12] で測定した結果である．端末 10 台で構成するアドホックネットワークにおけるスループットは約 36%低下した．本

表 4 スループット
Table 4 Throughput.

セキュアルーティング機能の有無	スループット (10^6bps)
なし	0.198
あり	0.126

処理にともなうパケットのデータ量増加は小さく，スループットへの影響はきわめて小さい．スループットの低下は認証，制御メッセージの改ざん検知のセキュリティ処理にともなうものであるため，高性能なプラットフォームにおいては，性能低下が軽減されることが期待できるが，低リソースな端末にむけては，スループット向上のためのさらなるセキュリティ処理の向上が必要である．

5. おわりに

アドホックネットワークの不正利用を防止することを目的に，アドホックネットワーク OLSR 向けセキュア・ルーティング方式を開発した．

直接通信可能な端末間で非対称鍵暗号による端末間認証を行い，不正な端末がルーティングテーブルに登録されることを防ぐ．その過程で対称鍵暗号系の鍵を共有し，処理が軽量な対称鍵暗号による制御メッセージの完全性検証を行うことで，定常時のセキュリティオーバヘッドが少ないセキュア・ルーティング方式を実現した．また，全端末がネットワークトポロジ情報の共有と同時に，共通の対称鍵暗号系のアドホック鍵を共有する仕組みを有することで，端末が移動した際には対称鍵暗号による端末認証を行い，ネットワークトポロジの変化にともなう認証のセキュリティオーバヘッドを，開発の結果 20%以下に低減させた．また，端末 10 台で構成するアドホックネットワークにおけるスループットは，提案方式の実装により，約 36%低下することを確認した．

今後は，アドホック鍵を活用したアプリケーションデータの改ざん検知や秘匿，セキュア・ルーティング方式のさらなる軽量化，あるいは鍵更新間隔などのパラメータの評価を進め，多様な自律分散型のネットワークシステムに共通的に適用できる方式に拡張する．

なお本開発では，リアル鍵，アドホック鍵を MAC 生成，暗号化，認証などの複数の目的に共通に用いたが，用途別に共有することも可能である．

また，アドホックネットワークにおける鍵管理の問題として，鍵の失効情報などをどのように扱っていくかは今後の課題である．

本研究は，独立行政法人情報処理推進機構（IPA）殿次世代ソフトウェア開発事業「セキュアなモバイル & AdHoc 通信・情報管理機能の開発」（H16 年度）によって実施した．また，議論いただいた宝木和夫博士，瀬戸洋一博士，松井進氏に感謝いたします．

参 考 文 献

1) Mobile ad hoc network (MANET). http://www.ietf.org/html.charters/manet-xharter.html
2) Optimized Link State Routing Protocol (OLSR). http://www.ietf.org/rfc/rfc3626.txt
3) Topology Dissemination Based on Reverse-Path Forwarding (TBRPF). http://www.ietf.org/rfc/rfc3684.txt
4) Ad hoc On-Demand Distance Vector (AODV) Routing. http://www.ietf.org/rfc/rfc3561.txt
5) The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt
6) Dynamic MANET On Demand (DYMO) Routing. http://www.ietf.org/internet-drafts/draft-ietf-manet-dymo-03.txt
7) IEEE Standard for Information technology, Telecommunications and information exchange between systems, Local and metropolitan area networks, Specific requirements, Part 11: Wireless LAN Medium Access Control and Physical Layer specifications Amendment 6: Medium Access Control Security Enhancements.
8) Raffo, D.: An Advanced Signature System for OLSR, *Proc. 2004 ACM Workshop on Security of Ad Hoc and Sensor Networks* (2004).
9) 安藤ほか：アドホックネットワークにおけるセキュアルーティング方式の開発，情報処理学会，*DICOMO2005* (2005).
10) Ishida: Secure Routing Functions for OLSR Protocol, *2nd OLSR Interop & Workshop* (2005).
11) Open SSL Project Homepage. http://www.infoscience.co.jp/technical/openssl/
12) Public Netperf Homepage. http://www.netperf.org
13) 長船ほか：アドホックネットワークプロトコルの評価，電子情報通信学会アクティブネットワークと応用技術時限研究会 (2002.10).

(平成 17 年 12 月 2 日受付)
(平成 18 年 5 月 9 日採録)

福澤 寧子（正会員）

1985 年日本女子大学家政学部家政理学科物理学系卒業．同年（株）日立製作所システム開発研究所入所．以来，ソフトウェアの生産性，情報セキュリティシステムの研究開発に従事．2004 年 4 月より横浜国立大学環境情報学府博士後期課程に在籍．現在，ITS 等のモバイルシステムにおけるセキュリティ技術の研究開発に従事．

石田 修一（正会員）

1997 年 3 月東京工業大学大学院総合理工学研究科物理情報工学専攻修士課程修了．同年 4 月（株）日立製作所システム開発研究所に入所．以来，情報セキュリティ技術の研究開発に従事．主に暗号モジュール利用システムや，ネットワークにおける認証技術の開発を行っている．

安藤英里子

2002 年 3 月九州大学大学院数理学府修士課程修了．同年 4 月（株）日立製作所システム開発研究所に入所．以来，アドホックネットワーク等のモバイルシステムにおける情報セキュリティ技術の研究開発に従事．

松本 勉（正会員）

1986 年 3 月東京大学大学院博士課程（電子工学）修了．工学博士．同年横浜国立大学工学部専任講師．現在，同大学大学院環境情報研究院教授．1981 年より，暗号・電子署名のアルゴリズムとプロトコル，ディジタル証拠性，耐タンパソフトウェア，情報ハイディング，ネットワークセキュリティ，認証方式，バイオメトリクス，人工物メトリクス等の各種情報セキュリティ技術の研究教育とその実応用に力を注ぐ．国際暗号学会 IACR 理事．CRYPTREC 暗号モジュール委員会委員長．電子情報通信学会より「情報セキュリティの基礎理論」への貢献に関して業績賞を受賞．