社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2004－MBL－30 (10)
2004／9／17

センサネットワークのマルチホップツリー構造に適したセキュアな受信確認方式

八百 健嗣* 川本 康貴* 松村 靖子* 福永 茂*

概要

無線マルチホップネットワークおいて，サーバからのブロードキャストメッセージに対するノードの受信確認に関して，(i)通信のオーバヘッドが大きい，(ii)受信確認自体に信憑性がない，という問題がある．本稿では，マルチホップツリー構造のセンサネットワークにおいて，受信確認を効率良くかつセキュアに収集する手法を提案する．提案手法は，ノードが情報の受信を認証子を生成することでサーバに証明し，各親ノードは全子ノードの受信確認を自身の認証子と結合・圧縮しながらサーバへ返信することにより，受信確認の通信オーバヘッドを抑え，また，不正なノードによる受信確認の偽造を困難にしている．ツリー型のネットワークモデルを与え，1ノードあたりの通信回数について提案手法を評価した結果，親ノードが各子ノードごとに受信確認をサーバへ返信する場合と比較して，30%弱の通信回数で実現できた．

Secure and efficient acknowledgment for multi-hop tree structures in sensor networks

Taketsugu Yao† Yasutaka Kawamoto† Yasuko Matsumura† Shigeru Fukunaga†

Abstract

In wireless multi-hop networks, there are two problems for the nodes to acknowledge the receipt of the broadcast message from the server; (i) high traffic overhead, and (ii) unreliable acknowledgements. In this paper, we propose a secure and efficient acknowledgment for multi-hop tree structures in sensor networks. In this method, the nodes generate MACs to acknowledge the receipt of the message, and each parent node aggregates and hashes the acknowledgements generated by its child nodes. Therefore, the traffic overhead of the acknowledgements is reduced and it is difficult for a malicious node to fabricate the acknowledgments. We showed an example of the tree structure network model, and evaluated the number of the communications of the acknowledgements a node. Consequently, our method could reduce about 70 % of the number of the communications compared with the case where the parent node relayed the acknowledgement from each child node.

1 はじめに

近年，無線通信機能を持つセンサを多数設置して設備の管理や環境の観測などに役立てるセンサネットワークシステムが提案されている．センサネットワークシステムは，システムを管理・制御するサーバと，低コストで構成される膨大な数のセンサノードから構成され，マルチホップ通信形態により情報をやりとりすることが想定される．

ここで，センサネットワークにおける情報発信の形態の1つに，サーバから全ノードに対する情報発信（ブロードキャスト）がある．無線マルチホップネットワークにおいて，ブロードキャストに対する受信確認を行うことは，ネットワークの通信オーバヘッドが大きくなるという問題から懸念されるのが一般的である．しかし，例えば，センサノードに搭載するソフトウェアのアップロードデータなど，サーバが全ノードに対して重要な情報を発信する時は，サーバはノードに正しいデータが届けられたことを確認できるほうが好ましい．また，サーバが受信確認を得ることで，サーバはネットワークの形状やノード構成に変化がないことを確認することも可能になる．

* 沖電気工業株式会社 研究開発本部 ユビキタスシステムラボラトリ

† Ubiquitous System Laboratory, Corporate of Research and Development Center, Oki Electric Industry Co., Ltd.

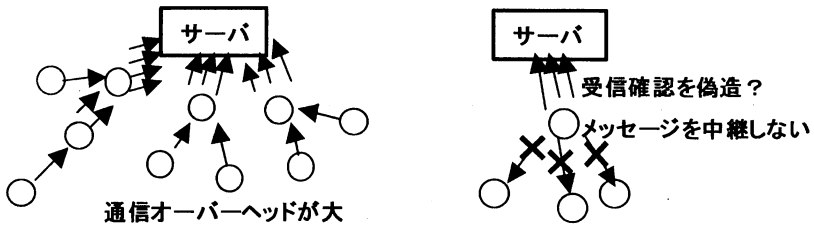


図1：ブロードキャストメッセージに対する受信確認を得ることの問題点

図1は，センサネットワークシステムにおいて，サーバがブロードキャストメッセージに対する受信確認を得ることの問題点を示した図である．サーバがブロードキャストメッセージの受信確認を行うには，次のような問題点がある．

- 膨大な数のノードが存在するセンサネットワークシステムにおいて，受信確認の返信は，ネットワークの通信オーバヘッドが大きい．（図1(a)）
- マルチホップ通信は，送信元の機器と送信先の機器との間に複数のノードが中継する形態をとるため，中継ノードからの受信確認メッセージに信憑性がない．（不正な中継ノードが受信確認を偽造している可能性がある．）（図1(b)）

本稿では，センサネットワークの一般的なネットワーク構造であるツリー構造において，サーバが発信したブロードキャストメッセージに対する受信確認を，効率良くかつセキュアに収集する方式について提案・検討した結果を述べる．

以下ではまず，上記問題を解決するための従来手法として，Per-Hop Hash という概念を用いた受信確認手法について説明し，続いてツリー型のネットワーク構造に適した提案手法について説明し，実際にツリー型のネットワークモデルを与えて比較・評価した結果を述べる．

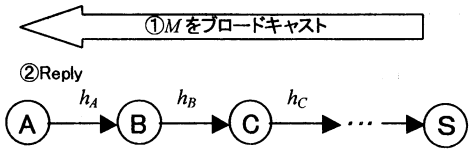
2 Per-Hop Hash による受信確認

2．1 Ariadne Route Discovery using MACs

Ariadne は，A.Perrig らにより提案されたセキュアルーティング方式である[1]．Ariadne においてセキュアにルートディスカバリーを実現する方式の一つに，Ariadne Route Discovery using MACs 方式がある．この方式は，任意のノードが他の全ノードと1対1の鍵を共有することを前提とする．

図2は Ariadne Route Discovery using MACs 方式の概念を用いてブロードキャストメッセージの受信確認を実現した場合の説明図である．図2において，ノードSはブロードキャストメッセージの発信ノードであり，ノードA，B，Cはブロードキャストメッセージの受信ノードである．

この方式は，ノードSからのブロードキャストメッセージを受信した各ノード（図2においてノードA，B，C，…）は；自身が正しいメッセージを受



証明情報

$h_A = \mathrm{H}(\mathrm{MAC}_{K_{SA}}(M))$.
$h_B = \mathrm{H}(\mathrm{MAC}_{K_{SB}}(M) \| h_A)$.
$h_C = \mathrm{H}(\mathrm{MAC}_{K_{SC}}(M) \| h_B)$.

M: ブロードキャストメッセージ，
$\mathrm{MAC}_{KXY}(\cdot)$: ノードXとYの共有鍵$K_{XY}$を用いたMAC生成アルゴリズム，
$String1 \| String2$: ビット列$String1$とビット列$String2$の連結，
$\mathrm{H}(\cdot)$: ハッシュ関数．

図2：Ariadneの概念を利用したブロードキャストメッセージの受信確認方法

信したことをノードSに証明するために，メッセージに対するMAC(Message Authentication Code)を，ノードSと各ノードが1対1で共有する鍵を用いて生成する．そして，ブロードキャストメッセージの最終到達ノード(図2においてノードA)から順に，生成したMACに対してハッシュ関数を施し，その出力値を証明情報としてサーバへ向けて返信してゆく．式(1)に，ノードAが生成する証明情報 h_A を示す．

$$h_A = \mathrm{H}(\mathrm{MAC}_{K_{SA}}(M)). \qquad (1)$$

ここで，M はブロードキャストメッセージ，$\mathrm{MAC}_{K_{XY}}(\cdot)$はノードXとノードYが1対1で共有する鍵 K_{XY}を用いたMAC生成アルゴリズム，H(・)はハッシュ関数である．ここで用いるMAC生成アルゴリズムは，例えば，AES暗号[2]等のブロック暗号を用いたCBC-MAC(Cipher Block Chaining - Message Authenticated Code)[3]であり，ハッシュ関数は例えば，MD5(Message Digest 5)[4]やSHA-1(Secure Hash Algorithm 1)[5]である．

以後，証明情報を受信するノード（図2においてノードB，C，…）は，自身の生成したMACと，前ホップのノードより受け取った証明情報をビット連結してハッシュ関数を施し，その出力値を自身の証明情報としてサーバへ向けて返信する．式(2)，(3)に，ノードB，Cそれぞれが生成する証明情報 h_B, h_C を示す．

$$h_B = \mathrm{H}(\mathrm{MAC}_{K_{SB}}(M) \| h_A). \qquad (2)$$

$$h_C = \mathrm{H}(\mathrm{MAC}_{K_{SC}}(M) \| h_B). \qquad (3)$$

ここで，‖はビット列の連結を示す．

ノードSは最終的に返信された証明情報と，各ノードと共有する鍵を用いて各ノードが生成するのと同じ手順で算出した証明情報とを比較することで，発信したメッセージの受信確認を得る．(ただし，ノードSはブロードキャストメッセージの受信確認がどのノードを通ってきたかを知る必要がある．)

この方式では，メッセージの受信証明情報は，複数のノードで重畳し，また，ノード毎にハッシュ（圧縮）するので，そのデータサイズは受信を証明するノード数に依存しない．よって，個々のノードがそれぞれ受信証明情報を送信する場合に比べて，通信のオーバヘッドが抑えられる．また，受信の証明は，受信証明の検証者と1対1で共有する鍵でMACを生成することにより実現するため，不正な中継ノードが受信確認情報を偽造することは，受信証明を重畳している1つまたは複数のノードのMAC生成鍵を暴露しないと困難である．

2．2 センサネットワークへ適用することへの課題

センサネットワークシステムは，システムを管理・制御するサーバと，低コストで構成される膨大な数のセンサノードから構成され，一般にツリー型のネットワーク構造をとる．2．1節で説明した受信確認方法（以下，従来手法と呼ぶ）をサーバのブロードキャストメッセージの受信確認に適用しようとする場合，ツリー型のネットワーク構造における各エンドノード（子ノードを持たないノード）がそれぞれサーバに対して受信確認を返信する形態となり，ルーターノード（子ノードを持つノード）は，自身の下流に存在するエンドノードの個数の受信確認を親ノードへ中継する必要がある．例えば，図3の場合では，サーバは，"Reply1"によりノードA，B，Dの受信を確認でき，"Reply2"によりノードA，

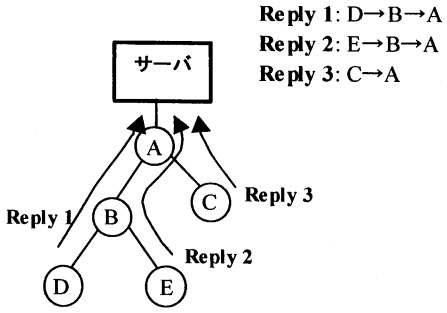


図3：従来手法を用いた受信確認の返信

B，Eの受信を確認でき，"Reply3"によりノードA，Cの受信を確認できるが，この場合ノードAとBは自身の受信確認が含まれる情報を複数回中継しており，効率が良くない．

以上より，従来手法をセンサネットワークシステムにおけるサーバのブロードキャストメッセージの受信確認にそのまま適用することは好ましくなく，センサネットワークシステムのツリー構造により適したメッセージ受信確認手法が必要である．

3 ツリー構造に適したセキュアな受信確認

この章では，ツリー構造のネットワークにおいて，サーバが発信したブロードキャストメッセージに対する受信確認を，効率良くかつセキュアに収集する方式について提案する．提案手法は，サーバは，自身が管理する全ノードと，認証鍵（各ノードがサーバと1対1で共有する鍵）と，ルート情報を共有することを前提とする．

3．1 ノードによる受信確認の生成

図4にノードによる受信確認の生成例を示す．図4において，Mはサーバが発信したブロードキャストメッセージを示し，K_A, K_B, K_C はそれぞれノードA，B，Cの認証鍵を示し，h_A, h_B, h_C はそれぞれノードA，B，Cの生成する受信確認を示す．ノードXの生成する受信確認 h_Xを式(4)に示す．

$$h_X = \mathrm{H}(\mathrm{MAC}_{K_X}(M)\{\|[\text{子ノードの受信確認}]\|\cdots\}) \tag{4}$$

ここで，K_XはノードXの認証鍵を示し，"{‖[子ノードの受信確認]‖…}"は，ノードXがルーターノードである場合に，ノードXの子ノードが生成する受信確認を示す．

各ノードは，サーバのブロードキャストメッセージに対する自身の受信証明情報として，受信したブロードキャストメッセージMに対するMACを，自身の認証鍵を用いて生成する．提案手法においてノードは，自身の親ノードと子ノードを示すノード情報テーブルを，サーバと共有している．自身がエンドノード（図4においてノードB，C）の場合は，生成したMACに対するハッシュ値を自身の受信確認情報として親ノードへ送信する．自身がルーターノード（図4においてノードA）の場合は，生成したMACと，自身の全子ノードから受け取った受信確認情報を連結させたものに対してハッシュ関数を施し，その出力値を自身の受信確認情報として親ノードに送信する．このように提案手法では，各親ノードは，自身が生成したMACと子ノードの受信確認とを重畳させたものを自身の受信確認として，サーバに返信することを特徴とする．

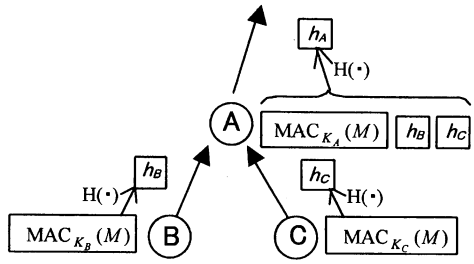


図4：ノードによる受信確認の生成例

3．2 サーバによる受信確認の算出

サーバは，自身が管理する全ノードと，認証鍵と，ルート情報を共有する．表1に，サーバが管理するノード情報の例を示す．ここで，ノードA，Bはルーターノードであり，ノードC，D，Eはエンドノードである．また，K_A, K_B, K_C, K_D, K_E はそれぞれ，ノードA，B，C，D，Eの認証鍵を示す．

表1：サーバが管理するノード情報の例

ノード	認証鍵	親ノード	子ノード
A	K_A	サーバ	B，C
B	K_B	A	D，E
C	K_C	A	—
D	K_D	B	—
E	K_E	B	—

サーバは，自身が発信したブロードキャストメッセージ M と，自身が管理するノード情報より，受信確認を次のように算出する．

- 全ノードに関して，ブロードキャストメッセージ M に対するMACを各ノードの認証鍵を用いてそれぞれ求める．
- エンドノードに該当するノード（表1においてノードC, D, Eに相当）に関して，求めたMACに対してハッシュ関数を施した結果をそのノードの生成する受信確認とする．
- ルーターノードに該当するノード（表1においてノードA, Bに相当）に関して，求めたMACと，そのノードの全子ノードの受信確認とを連結したビット列に対してハッシュ関数を施した結果をそのノードの生成する受信確認とする．

以上のように，サーバは，ノードから受信すると想定される受信確認を，ノードが生成するのと同じ手順で算出する．

3．2 提案手法の動作と特長

図5に提案手法の動作例を示す．提案手法は，サーバが管理下の全ノードに対してメッセージをブロードキャストし（図5において左図），そのメッセージに対してノードが3．1節の手順に従って受信確認を生成する過程と，サーバが3．2節の手順に従って受信確認を算出する過程と，サーバがノードから返信された受信確認と，算出した受信確認が一致するかどうかを検証する過程より成る（図5において右図）．ここで図5において，h_A, h_B, h_C, h_D, h_E はそれぞれ，ノードA, B, C, D, Eが生成する受信確認を示す．また，h'_Aは，サーバが自身の管理するノード情報より算出した受信確認を示す．

ノードによる受信の証明は，サーバとノードが共有する認証鍵でMACを生成することにより実現するため，不正な中継ノードが受信確認情報を偽造することは，受信証明を重畳している1つまたは複数のノードのMAC生成鍵を暴露しないと困難である．例えば，図5においてノードAが不正なノードであり，サーバからのブロードキャストメッセージを中継せず，受信確認を偽造しようとする場合は，ノードB,C,D,Eの認証鍵を暴露しないと正しい受信確認を偽造することは困難である．サーバは，ノードから返信された受信確認（図5において h_A）と，算出した受信確認（図5において h'_A）が一致するかどうかにより，以下のことがわかる．

- 発信したブロードキャストメッセージが自身の管理するノード群に正しく届けられたか，そうでないかがわかる．
- 管理するノード情報に変化がないか，そうでないかがわかる．（ネットワークの形状やノード構成に変化がないか，そうでないかがわかる．）

また，2．1節で説明した従来手法に比べて提案手法は，次のような点でツリー型のセンサネットワークシステムに適している．

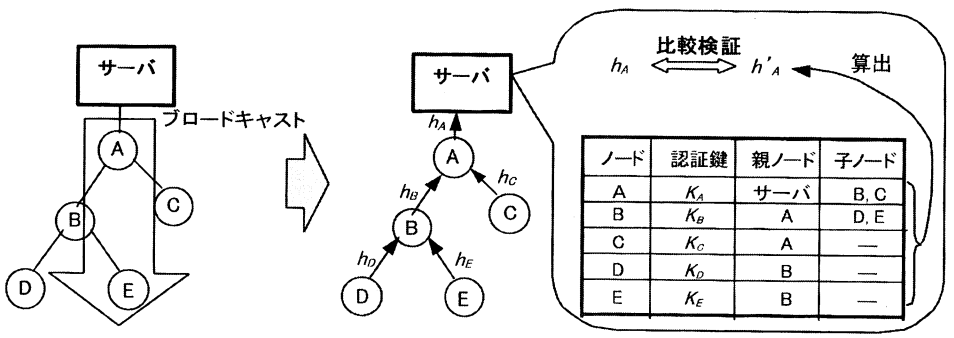


図5：提案手法の動作例

・ 受信確認は，各ルーターノードで全子ノードの受信確認を重畳しながら返信するため，各ルーターノードにおける受信確認の中継送信回数は1回のみである．

4 比較と評価

この章では，実際にツリー型のネットワークモデルを与えて，従来手法と提案手法における通信回数の比較と評価を行った結果を述べる．ここで，従来手法は2．1節で説明した受信確認方法であり，サーバのブロードキャストメッセージに対して，ツリー型のネットワーク構造における各エンドノードが受信確認を返信する方法である．

本稿で利用したツリー型ネットワークモデルを図6に示す．図6において，H (≧2)は木の高さ，d(1≦d ≦H)は木の深さ，m (≧2)は子ノードの数をそれぞれ示す．また，全てのエンドノードはHの深さをもち，全ての親ノードにはmの子ノードが接続しているものとする．ここで，深さdをもつノードの総数と，深さd以下のノードの総数と，深さdの1ノードに繋がるエンドノードの総数は，それぞれ次のように表される．

・ 深さdをもつノードの総数： m^{d-1}

・ 深さd以下のノードの総数：

$$1+m+\cdots+m^{d-1}=\frac{m^d-1}{m-1}$$

・ 深さdの1ノードに繋がるエンドノードの総数： m^{H-d} ただし，$d \neq H$

以上の結果を踏まえて，図6に示すネットワークモデルを用いて，従来手法と提案手法で送信回数と受信回数について比較した結果を表2と表3に示す．

表2：深さdの1ノードの送信回数と受信回数の比較

	従来手法	提案手法	従来手法／提案手法
送信回数	m^{H-d}	1	m^{H-d}
受信回数 ($d \neq H$)	m^{H-d}	m	m^{H-d-1}

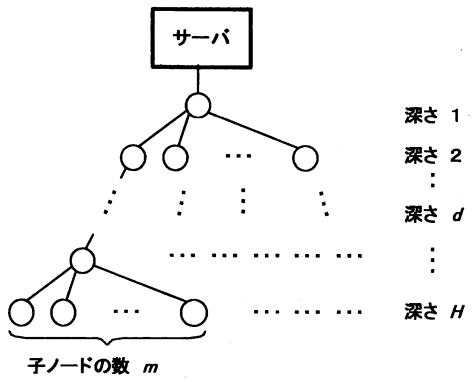


図6：ツリー型のネットワークモデル

表2は，深さdの1ノードに関して受信確認の送信回数と受信回数の比較を行った結果である．表2より，従来手法では，木の高さHに対する深さdが小さいノードほど，子ノードの数mのべき乗オーダーで送信回数と受信回数が大きくなるのに対して，提案手法では，木の高さHとノードの深さdに依存せず，送信回数は1回のみであり，受信回数は子ノードの数mとなる．ただし，エンドノードは受信確認を送信するのみで受信しないので，受信回数は0回である．

表3は全ノードが送受信する受信確認の全送信回数と全受信回数を全ノード数で割った，1ノードあたりの送信回数と受信回数に関して比較を行った結果である．表3内における従来手法と提案手法の送信回数の比を，木の高さHに対して，子ノードの数mが2,3,4,5の場合について求めた結果を図7に，また受信回数の比について同じように求めた結果を

表3：1ノードあたりの送信回数と受信回数の比較

	従来手法	提案手法	従来手法／提案手法
送信回数	$\frac{Hm^{H-1}(m-1)}{m^H-1}$	1	$\frac{Hm^{H-1}(m-1)}{m^H-1}$
受信回数	$\frac{(H-1)m^{H-1}(m-1)}{m^H-1}$	$\frac{m^H-m}{m^H-1}$	$\frac{(H-1)m^{H-1}(m-1)}{m^H-m}$

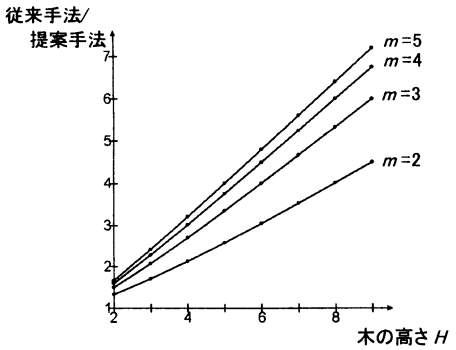


図7: 1ノードあたりの送信回数の比較

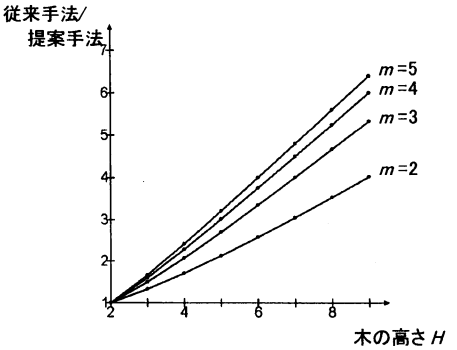


図8: 1ノードあたりの受信回数の比較

図8に示す．図7，図8より，1ノードあたりの送受信回数は，木の高さ H と子ノードの数 m が大きいほどその差も大きくなることがわかる．例えば，木の高さ H を3，子ノードの数 m を3と設定した場合，1ノードあたりの送信回数の比，受信回数の比はそれぞれ，2.1，1.5となり，1ノードあたりの送受信回数の比は合計 3.6 になる．送受信で費やす消費電力量は単純に送受信回数に比例すると想定すると，送受信で費やす電力消費量だけに着目した場合は，提案手法は従来手法と比べてノードの寿命が3.6倍になることを示している．

5 まとめ

以上，センサネットワークのツリー構造において，サーバが発信したブロードキャストメッセージに対する受信確認を，効率良くかつセキュアに収集する方式について提案し，実際にツリー型のネットワークモデルを与えて，比較・評価した結果を述べた．

今後は，パケットロス等によるブロードキャストデータ不達ノードの対応や計算量を含めた消費電力の比較，また，提案手法の実機への実装による評価が課題となる．

参考文献

[1] Yih-Chun Hu, Adrian Perrig, David B. Johnson, "Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks," *MobiCom'02*, Atlanta, Georgia, USA, September 2002.

[2] NIST FIPS PUB 197, "Advanced Encryption Standard (AES)," November 2001.

[3] NIST FIPS PUB 113, "Computer Data Authentication," May 1985.

[4] R. Rivest, "The MD5 Message Digest Algorithm", RFC 1321, April 1992.

[5] D. Eastlake, 3rd and P. Jones, "US Secure Hash Algorithm 1 (SHA1)," RFC3174, September 2001.