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

2007－DPS－130 (22)
2007－CSEC－ 36 (22)
2007／3／1

# ユーザ失効を考慮した匿名IEEE802.1X認証の実装

三木康平† 中西透†
川島潤† 舩曳信生†

インターネット接続のためのアクセスポイントにおける認証方式としてIEEE802.1X認証が利用されているが，現在の方式ではモバイルホストのプライバシー情報が通信事業者に漏洩する問題がある．我々はこの問題を解決するために，グループ署名を用いた匿名認証方式を提案し，実装してきた．しかし，この匿名認証方式ではユーザ失効が考慮されていないという問題がある．そこで本稿では，ユーザ失効可能なグループ署名を利用することにより，ユーザ失効機能をもつ匿名IEEE802.1X認証の実装を行い，その評価を行う．

# An Implementation of an Anonymous IEEE802.1X Authentication with User Revocation

Kouhei Miki,† Toru Nakanishi,† Jun Kawashima† and Nobuo Funabiki†

The IEEE802.1X authentication is used well as an authentication protocol in the Access Point for the Internet connection. There is a problem that mobile host's privacy information is leaked to the Internet service provider in this protocol. To solve this problem, we have proposed and implemented an anonymous authentication using the group signature. However, there is a problem that the user revocation is not considered. In this paper, an anonymous IEEE802.1X authentication with a user revocation is implemented by using a group signature with a user revocation function, and the authentication time is evaluated.

## 1. はじめに

近年，インターネットに接続可能なモバイル端末の普及とともに，アクセスポイント（以下，AP）と呼ばれる無線を利用したインターネットの接続サービスが，インターネット接続事業者（以下，ISP）により提供されている．APの利用者は，あらかじめISPと契約してその利用権を得ることで，インターネットに接続できる．ISPは契約した利用者のみに接続サービスを提供するために，ユーザ認証を行い本人確認をする．この認証規格の一つにIEEE802.1Xがあり[1]，現在，無線におけるAPでの認証に利用されている．

APにおけるユーザ認証はプライバシー問題を起こし得る．これは，その利用者がどこの場所にあるAPからいつ接続したか，また，インターネットでどこに接続したかを示す履歴をISPが把握できるためである．もし，これらの履歴情報を他に漏洩されたとしても，ユーザは関知できない．さらに，ISPはこれら利用者のプライバシー情報が漏洩しないように管理する必要があるが，管理すべき情報が大量となるため管理は繁雑になる．

このプライバシー問題は，APに接続するユーザの認証においてISPに利用者を特定できる情報が渡ることが原因である．この問題を回避するため，我々はグループ署名を用いた匿名IEEE802.1X認証を提案・実装してきた[2]．しかし，この従来方式ではユーザ失効が考慮されていない．匿名認証では利用者が誰であるか特定することが不可能なため，ユーザ名を通じてユーザ失効を行うような単純な失効処理はできない．失効処理ができない場合，契約した利用者は無期限で無線LANへの接続が可能となってしまい問題となる．そこで本稿では，ユーザ失効可能なグループ署名を用いることによりユーザ失効機能の実装を行う．また，認証時間の評価を行い，その有効性を示す．

以下，本稿の章構成を示す．2章ではIEEE802.1X

† 岡山大学大学院自然科学研究科
Graduate School of Natural Science and Technology, Okayama University, E-mail:{miki, kawasima}@sec.cne.okayama-u.ac.jp, {nakanisi,funabiki}@cne.okayama-u.ac.jp

認証について示す．3章では匿名認証を実現する従来方式とその問題点について示す．4章ではユーザ失効を考慮した認証方式を提案する．5章では提案方式におけるユーザ失効，認証部分の実装方法と内部処理について示す．6章では認証時間について示す．最後に7章で本稿のまとめを示す．

## 2. IEEE802.1X 認証

IEEE802.1X とは，LAN 内でユーザ認証を行うための方式を定めた規格である．IEEE802.1X では，EAP(Extensible Authentication Protocol) と呼ばれる通信プロトコルを認証に採用しており，「ユーザ ID・パスワード」による認証や電子証明書による認証など，さまざまな認証方式に対応している．

IEEE802.1X 認証では RADIUS(Remote Authentication Dial In User Service) と呼ばれる認証サーバ（以下，認証サーバ）が必要である．一方，認証を受ける端末にはサプリカントと呼ばれる IEEE802.1X 認証に準拠した認証を実現するためのソフトウェアが必要となる．サプリカントと認証サーバ間の通信を中継する機器として，IEEE802.1X に対応した AP やスイッチを設置する必要がある．

サプリカント・AP 間の通信は EAP プロトコルで行い，AP・認証サーバ間の通信は RADIUS プロトコルで行う．AP は EAP プロトコルと RADIUS プロトコルの相互変換（パケットデータの詰め替え）を行い，サプリカント・認証サーバ間の通信を中継するためのプロクシとして動作する．

未認証の端末は AP より外部のネットワークには接続できない．認証サーバが端末の認証を完了すると，AP へ接続許可のパケットを送り，端末は AP より外部のネットワークに接続が可能となる．

EAP には各種の認証方式が定義されている．主として以下の 3 方式がある．

- EAP-MD5
  ID とパスワードを MD5 ハッシュで暗号化して送信し，ユーザ認証を行う．
- EAP-TLS
  RADIUS 認証サーバと認証端末の双方で電子証明書を交換し，電子証明書によって相互認証を行う．
- EAP-TTLS
  認証端末は RADIUS 認証サーバから提供された電子証明書を利用してサーバ認証を行い，認証サーバはパスワードを用いてユーザ認証を行う．

EAP-TTLS の処理の流れを図 1 に示す．従来方式は簡単化のため既存の EAP-MD5 方式を基に実装されている[2]．この認証方式では認証サーバの認証（サーバ認証）が行われないため，悪意を持つ第三者による認証サーバなりすまし問題が起こりえる．そのため，本稿では，EAP-MD5 方式から EAP-TTLS 方式への移行を行い，それを基に実装を行う．

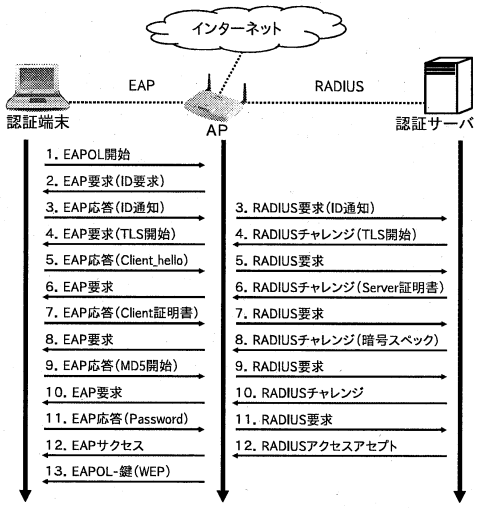


図 1　EAP-TTLS のプロトコルフロー

## 3. 従来方式とその問題

以下，従来方式とその問題点について示す．

### 3.1 グループ署名を用いた匿名認証

従来方式ではグループ署名を用いている．グループ署名では，グループ管理者（GM）と呼ばれる特別な機関を必要とする．GM は，あるユーザがグループに加入し，メンバーとなることを許可する権限を持つ唯一の機関である．ユーザは GM からグループ証明書を発行してもらうことでグループ署名を作成可能となり，その署名を用いて匿名で正規ユーザであることを証明できる．ただし，指定された追跡機関の検証者のみ，不正が発生した際に署名の作成者を特定できる（追跡可能性）．

匿名 IEEE802.1X 認証のモデルを図 2 に示す．

- 認証端末
  ユーザのモバイル端末．認証時にはグループ署名の生成を行う．
- AP
  IEEE802.1X 認証に対応している無線 LAN のアクセスポイントである．
- 認証サーバ
  RADIUS サーバ．認証時にはグループ署名の検

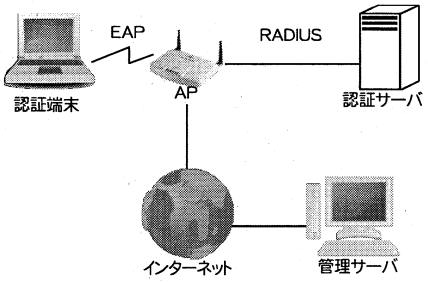


**図 2** 匿名 IEEE802.1X 認証のモデル

証を行う．

- 管理サーバ

  GM．ユーザの個人情報を保持し，ユーザがグループに加入しメンバーとなることを許可する．また，追跡機関を兼ねており，不正発生時にはグループ署名の匿名を開示する．

従来方式では，上記のモデルにおいて以下の各処理を行う．

**ユーザ登録** ：

予めユーザは GM である管理サーバにユーザの情報を登録しておく．ここで登録する情報はユーザ個人を特定可能な情報である．登録後，管理サーバからグループ証明書の発行を受ける．この証明書を用いてグループ署名の作成が可能となる．

**ユーザ認証** ：

ユーザ登録において受け取ったグループ証明書を用いてグループ署名の作成を行う．このグループ署名を認証サーバに送信して認証を行う．認証サーバにおいて，認証端末から受信したグループ署名の検証を行う．検証が成功した場合，無線 LAN への接続を許可する．

**不正を行ったユーザの追跡** ：

不正を行ったユーザとは，正規の認証を終えて無線 LAN を利用しているユーザのうち，不正アクセス等のインターネット犯罪を行ったユーザの事である．この不正を行ったユーザにはグループ署名の性質である追跡可能性を利用する．認証サーバは認証端末が認証時に用いたグループ署名をログとして保存する．このグループ署名を管理サーバに送信して，匿名の無効化を要求する．

### 3.2 ユーザ失効の問題

匿名認証ではユーザが誰であるか特定することが原則として不可能であり，ユーザ名を通じてユーザ失効を行うような単純な失効処理はできない．したがって，匿名ユーザの失効を行うことは容易ではない．従来方式ではユーザ失効を考慮していないため，一度ユーザ登録したユーザは無期限で無線 LAN への接続が可能となる．このため，不正を行ったユーザ，または契約期限の切れたユーザに対しての無線 LAN サービス停止も行えない．

## 4. ユーザ失効を考慮した匿名 IEEE802.1X 認証の提案

提案する認証方式におけるモデルは従来のモデル図 2 と同様である．従来方式に対して，ユーザ失効可能なように以下の処理の追加変更を行う．

- ユーザ失効
- ユーザ認証

ユーザ失効可能なグループ署名方式が様々提案されてきているが，いずれの方式でも失効発生時に GM がユーザ失効情報を生成し，署名者や検証者に配布する．そして最新の失効情報に基づいて署名作成・検証が行われる．したがって，グループ署名を用いてユーザ失効を行うためには GM である管理サーバが作成する失効情報をユーザと認証サーバに送信しなくてはならない．本稿では，ユーザ失効時に管理サーバから認証サーバへ失効情報の送信を行い，ユーザ認証時に認証サーバからユーザへ失効情報を送信する．以下，各処理について示す．

### 4.1 ユーザ失効

匿名ユーザの無線 LAN サービスを停止するためのユーザ失効処理について示す．処理の流れを図 3 に示す．管理サーバは，ユーザ失効が起こるたびにユーザ失効情報の更新を行う．更新後，全ての認証サーバに最新の失効情報を送信する．認証サーバでは最新の失効情報を認証に利用することでユーザのサービス停止が可能となる．

### 4.2 ユーザ認証

ユーザが無線 LAN サービスを利用するための認証処理について示す．処理の流れを図 4 に示す．

① ユーザ失効情報の送信

認証サーバは失効されたユーザを接続不可能にするためのユーザ失効情報を保持している．この失効情報をユーザの認証端末に送信する．

② グループ署名の作成と送信

3.1 節のユーザ登録において受け取ったグループ証明書とユーザ失効情報を用いてグループ署名の作成を行う．このグループ署名を認証サーバに送信して認証を行う．

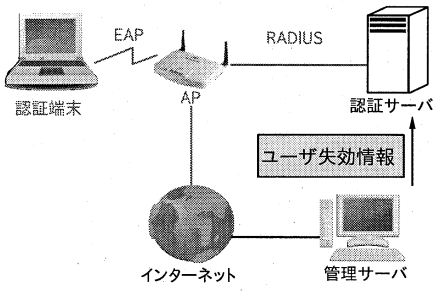


**図 3** 提案方式の処理：ユーザ失効

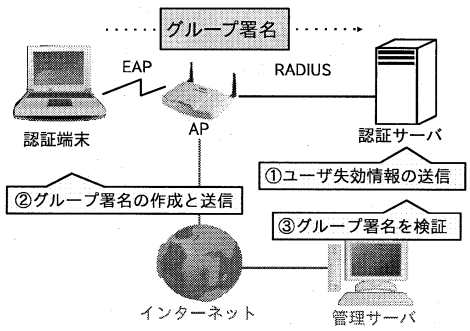


**図 4** 提案方式の処理：ユーザ認証

③ グループ署名を検証

認証サーバにおいて，認証端末から受信したグループ署名の検証を行う．正しく検証が行われた場合，無線 LAN への接続を許可する．もし失効したユーザが認証を要求した場合，グループ署名の検証に失敗するため，その接続は許可されない．

認証サーバには，設定された時間ごとにユーザに再認証を求める機能が備わっている．その機能を用い再認証を行うことで，一度認証に成功し無線 LAN への接続を許可されたユーザに対しても失効が行える．

## 5. 実　　装

### 5.1 実装したグループ署名

実装では，ユーザ失効可能であり高速に署名作成・検証が可能な文献の方式[3]（方式 1）を用いた．このグループ署名は以下の特徴がある．

- RSA 暗号ベースである
- ユーザごとにユニークな素数情報を割り当てたグループ証明書を発行し，この素数情報に基づいてユーザの失効処理を実現する．この素数情報がユーザ失効情報となる
- 既存のグループ署名の中では比較的高速に署名の作成・検証が行える

実装では，鍵長は 2048bit とした．

### 5.2 ユーザ失効

管理サーバから認証サーバへのユーザ失効情報の送信には，UNISON と呼ばれるファイル同期ソフトを流用した[4]．この UNISON は以下の特徴がある．

- rsync アルゴリズムを用いて高速な同期を行う
- 前回同期時の状況を記録しており，増減したファイルを自動的に割り出して，コピーや削除を行う
- SSH を利用したセキュアな同期も可能

管理サーバでユーザ失効情報の更新があるたびに認証サーバに失効情報を送信し，失効情報の同期を取る．これにより，認証サーバは常に最新の失効情報を保持することができる．

### 5.3 ユーザ認証

図 4 における①，②及び③の認証処理については，既存の認証方式である EAP-TTLS を基に拡張し，新たに EAP-TTLS/GS として実装した．

#### 5.3.1 EAP-TTLS/GS のパケットシーケンス

以下に，EAP-TTLS/GS の処理の流れを示す．EAP-TTLS からの変更点は図 1 の STEP10，STEP11，STEP12 の処理であり，他の処理は EAP-TTLS と同じである．

**STEP10** ：EAP 要求 / RADIUS チャレンジ

認証サーバからユーザの認証端末へ 128bit の乱数とユーザ失効情報を送信する (①の処理)．EAP パケットサイズは Ethernet の MTU(1500Byte) に依存しており，一つの EAP パケットに入るデータの大きさは約 1460Byte となっている．ユーザ失効情報がパケットサイズを超える場合，分割送信を行う．送信する乱数は，認証サーバで保存する．

**STEP11** ：EAP 応答 / RADIUS 要求

乱数とユーザ失効情報の受信が終わった後に，認証端末はグループ署名の作成を行う (②の処理)．このとき，署名のメッセージとして認証サーバから受信した乱数を用いる．グループ署名作成後，認証端末から認証サーバへグループ署名の送信を行う．ユーザ失効情報と同様に，パケットサイズを超える場合，グループ署名の分割送信を行う．

**STEP12** ：EAP 成功 / RADIUS アクセス許可

認証サーバは認証端末から受信したグループ署名を，STEP10 の処理で保存した乱数をメッセージとして検証する (③の処理)．検証が成功した場合は認証サーバは AP に Access-Accept と呼ばれ

| Code (8bit) | Identifier (8bit) | Length (16bit) |
|---|---|---|
| Type (8bit) | Type-Data ... (可変長bit) | |

[Code] : 要求 (1)　　[Length] : パケット長 (byte)
[Identifier] : 2　　[Type] : EAP-TTLS/GSを定義 (610)

Type-Data

| Finish_Message (8bit) | Value_Size (8bit) | Value_Data (128bit) |
|---|---|---|
| Revoke_Size (16bit) | | Revoke_Data (可変長bit) |

[Finish_Message] : Value_Data,Revoke_Data送信完了時 1,else 0
[Value_Size] : Value_Dataの長さ (128(bit))
[Value_Data] : 128(bit)の乱数値
[Revoke_Size] : Revoke_Dataの長さ (byte)
[Revoke_Data] : ユーザ失効情報

**図 5**　STEP10 : EAP 要求パケットフォーマット

| Code (8bit) | Identifier (8bit) | Length (16bit) |
|---|---|---|
| Type (8bit) | Type-Data ... (可変長bit) | |

[Code] : 応答 (2)　　[Length] : パケット長 (byte)
[Identifier] : 2　　[Type] : EAP-TTLS/GSを定義 (610)

Type-Data

| Value_Size (16bit) | | GS_Version (8bit) | Finish_Message (8bit) |
|---|---|---|---|
| Param_Num (8bit) | Param_Type (8bit) | Odd_Flag (8bit) | Param_Size (16bit) |
| Param_Data (可変長bit) | ... | | |

[Value_Size] : Type_Dataの長さ
[GS_Version] : グループ署名の種類 (1)
[Finish_Message] : 全てのParam_Data送信完了時 1,else 0
[Param_Num] : グループ署名のパラメータ数 (14)
[Param_Type] : パラメータの変数名
[Odd_Flag] : Param_Sizeが8の倍数bitの場合 0,else 1
[Param_Size] : パラメータ値の長さ (byte)
[Param_Data] : パラメータ値

**図 6**　STEP11 : EAP 応答パケットフォーマット

るコードの RADIUS パケットの送信を行い，失敗した場合は Access-Reject と呼ばれるコードの RADIUS パケットの送信を行う．

### 5.3.2 EAP-TTLS/GS のパケットフォーマット

独自に定義した STEP10 と STEP11 の EAP パケットフォーマットを図 5，図 6 に示す．Type により認証方式の情報種類を規定し，Type-Data に認証情報に対応したデータを格納する．認証方式が EAP-TTLS/GS であることを表すために，Type に固有の値「610(未定義の値を選択)」を定義し，さらに認証端末と認証サーバ間でユーザ失効情報とグループ署名を送受信できるように図 5，図 6 のように Type-Data のフォーマットを定義した．EAP 応答パケットフィールドの Param_Num はグループ署名のパラメータ数を表す．グループ署名のパラメータとは，今回用いる文献の方式[3]（方式 1）において $c, u, U_1, U_2, U_3, z_x, z_d, z_e, z_R, z_k, z_\alpha, z_\beta, z_\gamma, z_\delta$ の 14 個の変数であり，今回は 14 の値が入る．Param_Type はパラメータの識別番号である．実際のパラメータ値は Param_Data に入る．今回は上記の順に 1 から 14 の値が入る．なお，Param_Type，Odd_Flag，Param_Size，Param_Data の各フィールドはパラメータ毎に存在する．

**表 1**　認証端末の実装環境

| | |
|---|---|
| 認証端末 | Xsupplicant-1.2.8 (+libdnet-1.10, Openssl-0.9.8d, gmp-4.2.1) |
| OS | GentooLinux kernel-2.6.16-gentoo-r7 |
| CPU | PentiumM 1.7GHz |
| Mem | 512MB |
| 無線 LAN | Intel(R) PRO/Wireless LAN 2100 IEEE802.11b 11Mbps にて接続 |

**表 2**　認証サーバの実装環境

| | |
|---|---|
| 認証サーバ | FreeRADIUS-1.1.3 (+Openssl-0.9.8a, gmp-4.1.4-11) |
| OS | UbuntuLinux6.06 kernel 2.6.15-27-386 |
| CPU | AMD Sempron(tm) 2600+ 1.6GHz |
| Mem | 700MB |
| NIC | Intel(R) 82547EI.　100Mbps にて接続 |

### 5.4 実装環境

認証端末，認証サーバの実装環境をそれぞれ表 1，表 2 に示す．ネットワーク環境を図 7 に示す．また，AP については BUFFALO WLM2-L11G を用いた．提案方式はオープンソースのソフトウェア Xsupplicant 及び FreeRADIUS を基に実装した．また認証端末には Xsupplicant へ EAP-TTLS/GS のパケットフォーマットに基づく通信処理とグループ署名の作成処理を組み込んだ．認証サーバには FreeRADIUS へ EAP-TTLS/GS のパケットフォーマットに基づく通信処理とグループ署名の検証処理を組み込んだ．更に認証サーバには，管理サーバからの失効情報を UNISON を用いて同期する処理も組み込んだ[4]．

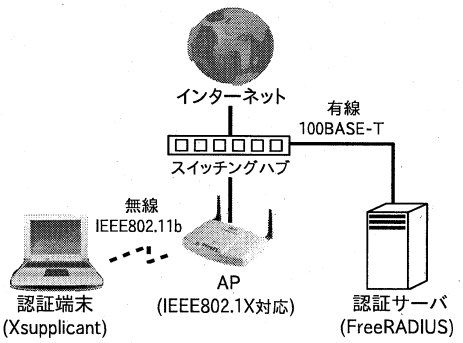


**図 7**　提案方式の実験環境

## 6. 実行結果と評価

本研究の提案方式である EAP-TTLS/GS の認証時間を図 8 に示す．また，認証時間のうちグループ署名作成・検証の合計時間を除いた通信時間とグループ署名作成・検証の合計時間を図 9 に示す．更に認証で送受信される通信データ量（ユーザ失効情報とグループ署名の合計）を図 10 に示す．比較のために従来方式[2]の認証時間，通信時間，署名作成・検証時間，通信データ量も各図に示す．

従来方式では，利用可能なユーザ数に関係なく認証時間が約 1.65 秒，通信時間が約 1.27 秒，署名作成・検証時間が約 0.38 秒，通信データ量が 1283byte と一定である．提案方式ではユーザ数が 100 人の場合，認証時間は約 2.51 秒，署名作成・検証時間は約 1.08 秒，通信データ量は約 1802byte であった．従来方式と比べて認証時間は増加しているが，両方式に大きな差は無く，実用的である．ユーザ数が 500 人の場合，認証時間は約 4.74 秒，署名作成・検証時間は約 2.27 秒，通信データ量は約 5067byte であり，従来方式に比べ認証時間が約 3 倍，通信データ量が約 4 倍となっている．更にユーザ数が 1500 人の場合，認証時間は約 10.26 秒，署名作成・検証時間は約 6.23 秒，通信データ量は約 13443byte であり，認証時間には多くの時間を要する．以上のことから，ユーザ数が大規模になるにつれ実用的な時間での認証は困難となってくる．これは，利用したユーザ失効可能なグループ署名の署名作成・検証時間及び通信データ量がユーザ数に依存するためである．その解決が今後の課題となる．

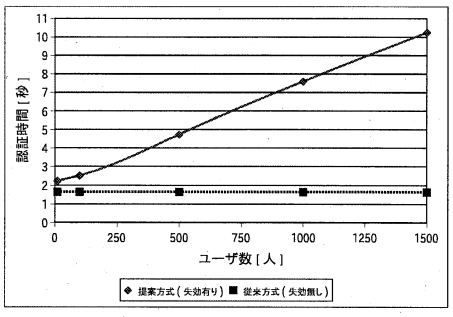


**図 8** 認証時間の比較

## 7. ま と め

本稿では，ユーザ失効を考慮した匿名 IEEE802.1X 認証の実装を行った．提案方式では，認証サーバのユーザ失効情報を管理サーバが更新し，認証時に認証サーバが失効情報を認証端末に送信する．それを基にグループ署名を作成し検証することでユーザ失効を実現した．また，提案方式の認証時間を測定し従来方式との比較を行った．認証時間に関しては，ユーザ数 100 人程度の小規模なグループの場合，実用的な時間で認証可能であることがわかった．今後の課題として，より大規模なグループ数に対しても高速に認証可能な方式への改良が挙げられる．

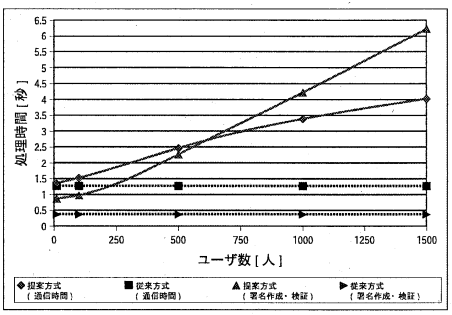


**図 9** 通信時間及び署名作成・検証時間の比較

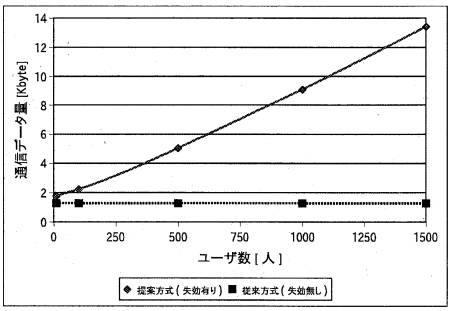


**図 10** 通信データ量の比較

## 参 考 文 献


1) Mattbew S. Gast 著，水野忠則 他訳，“802.11 Wireless Networks，” O'REILLY, 2004.
2) 高橋秀郎，川島潤，中西透，舩曳信生，“モバイルホストのプライバシーを秘匿する IEEE802.1X 認証の提案と実装，” 2006 年暗号と情報セキュリティシンポジウム（SCIS2006），2D4-3，Jan，2006.
3) 濱田直人，中西透，舩曳信生，“所属無効化可能なグループ署名方式の素数情報を用いた高速化とその実装，” 情報セキュリティ研究会（ISEC），pp.47-54，Jan,2006-12-13.
4) unison [Online], Available: http://www.cis.upenn.edu/ bcpierce/unison/