情報処理学会論文誌　Vol. 52　No. 9　2876–2881 (Sep. 2011)

推薦論文

サーバなりすまし攻撃に対し安全な RFID セキュリティ方式

丹羽弘和[†1]　高木剛[†2]　高橋修[†1]

RFID システムにおいて，サーバはタグから受け取った識別子と一致する値を検索してタグを識別する．したがって，サーバがタグを識別するためには，互いが同期していなければならない．しかし，SecureComm 2005 において，攻撃者がタグとサーバ間の送信データを改竄して同期を崩す攻撃が提案された．さらに UBICOMM 2008 において，攻撃者が正規のサーバになりすまし，タグの秘匿情報を更新することで同期を崩す攻撃が提案された．非同期状態のとき，サーバはタグを識別することができないため，タグの認証に失敗する．本稿では，従来のセキュリティ要件（識別不可能性，追跡不可能性，なりすまし攻撃耐性）を満たしつつ，データ同期を保つ方式を提案する．提案方式では，サーバに秘密鍵を格納し，タグに秘密鍵を 1 方向性ハッシュ関数で暗号化した値を格納する手法を用いる．仮に攻撃者がタグの秘匿情報を入手しても，1 方向性ハッシュ関数で暗号化されているため，入手したタグの秘匿情報からサーバの秘密鍵を計算できない．したがって，提案方式は攻撃者のサーバなりすまし攻撃を防ぐことができる．また，従来のセキュリティ要件も満たしている．

An RFID Protocol Secure against Server Impersonation Attacks

Hirokazu Niwa,[†1] Tsuyoshi Takagi[†2] and Osamu Takahashi[†1]

RFID systems identify a tag by searching for a synchronized identifier in a server. Therefore, values that are used to authenticate the RFID must be synchronized. A type of attack in which an adversary alters a value that is sent between a tag and a server to break synchronization was described at SecureComm 2005. In addition, attacks that desynchronize a tag and a server and attacks that impersonate a server by using the data stored in a tag were discussed at UBICOMM 2008. If a tag and a server become desynchronized, the server cannot identify the tag. The reasons for vulnerability to attacks that break synchronization in previous RFID security protocols are highlighted. To address this type of vulnerability, a method was developed for resynchronizing tags and servers, and a protocol using this method was implemented. The protocol meets important security requirements (indistinguishability, backward untraceability, and resistance against tag spoofing) for RFID systems and provides data synchronization.

1. はじめに

RFID は，無線通信上での自動識別を可能にする技術であり，タグとバックエンドで管理するサーバで構成されている．RFID は，従来のバーコード技術と比較して，自動認証が可能，複数のタグを同時に読み取ることが可能，通信可能距離が長いなどの利点がある[8),12)]．しかし，タグは計算資源が乏しいため従来の暗号技術（SSL など）を用いることが難しい．そのため，攻撃者による盗聴，改竄，なりすましなどの脅威が存在する．近年，安全な RFID システムを構成するために，以下のセキュリティ要件が提案された．

2003 年，Ohkubo らがタグの出力から特定のタグを識別できないという性質（識別不可能性）と，現在のタグの秘匿情報が漏洩しても所有者の過去の行動を追跡できないという性質（追跡不可能性）を RFID Privacy Workshop 2003 において提案した[6)]．さらに 2005 年，Rhee らが SPC 2005 にて，なりすまし攻撃に対して安全である性質（なりすまし攻撃耐性）を提案した[7)]．これらのセキュリティ要件を満たす方法の 1 つとして，認証ごとにタグとサーバが格納している識別子（*ID*）や秘密鍵などを更新する方法が有効であると知られている[1),2),4)–7),10),11)]．

しかし，サーバがタグを識別するためには，互いの秘匿情報（*ID* など）が同期していなければならない．本稿では，特にデータ同期に注目した方式を考察する．

SecureComm 2005 において，Dimitriou はタグとサーバ間の送信データを改竄して同期を崩す攻撃を提案した[2)]．この攻撃に対応するために，2008 年，Lim らが ISPEC 2008 にて，タグとサーバに同期化された鍵を格納する方式を提案した[5)]．さらに同じく 2008 年，

†1 公立はこだて未来大学大学院システム情報科学研究科
Graduate School of Systems Information Science, Future University Hakodate
†2 九州大学マス・フォア・インダストリ研究所
Institute of Mathematics for Industry, Kyushu University
本論文の内容は 2009 年 10 月のコンピュータセキュリティシンポジウム 2009（CSS2009）にて報告され，コンピュータセキュリティ研究会主査により情報処理学会論文誌ジャーナルへの掲載が推薦された論文である．

© 2011 Information Processing Society of Japan

Canard らが RFIDsec 2008 において，ハッシュチェイン方式を用いて同期を保つ方式を提案した[1]．しかし，UBICOMM 2008 において Song がサーバなりすまし攻撃を提案した[9]．これは，攻撃者がタグの格納している秘匿情報を入手し，正規のサーバになりすまして特定の正規タグが格納する秘匿情報を更新する攻撃である．しかし，攻撃者によって更新された正規タグに対し，正規サーバの秘匿情報は更新されないため，正規サーバと正規タグの同期が崩れてしまう．Lim や Canard らの方式は，サーバなりすまし攻撃に対して脆弱だった．

本稿では，識別不可能性，追跡不可能性，なりすまし攻撃耐性を満たし，かつ，サーバとタグのデータ同期を保つプロトコルを提案する．提案方式では，サーバに秘密鍵を格納し，タグに秘密鍵を 1 方向性ハッシュ関数で暗号化した値を格納する．仮に攻撃者がタグの秘匿情報を入手しても，タグの秘匿情報からサーバの秘密鍵を計算できないので，攻撃者は正規サーバになりすますことができない．また，タグとサーバが非同期状態ならば秘密鍵を用いてタグを識別し，鍵と対応している識別子（*ID*）を再同期化することで，サーバとタグの同期を保つことができる．従来のセキュリティ要件に関しては，認証ごとに秘匿情報を 1 方向性ハッシュ関数を用いて更新することで識別不可能性と追跡不可能性を満たし，乱数を暗号変換の計算に用いることでなりすまし攻撃耐性を満たす．

本稿の 2 章では，本稿で取り扱う RFID システムのモデル概要とセキュリティ要件について述べる．3 章では，従来方式の評価を行い，サーバなりすまし攻撃に対して脆弱であることを示す．4 章では，提案プロトコルの説明を行い，セキュリティ評価を行う．5 章では，本稿のまとめを述べる．

2. RFID セキュリティシステム

本章では，本稿で扱う RFID の認証プロセスについて述べる．

2.1 RFID の構成

RFID システムは，タグとバックエンドで管理しているサーバからなる．タグは，固有の *ID* を初期値としてメモリに格納している．サーバは，管理している全タグごとの *ID* とタグを取り付けている商品の情報（値段，商品名など）を対応付けて管理している．タグとサーバは認証ごとに同じアルゴリズムを用いて *ID* を固有の値に更新する．RFID システムにおける認証の流れは，次のとおりである；(1) サーバからの応答要求信号（*Query*）に対し，(2) 受信したタグが格納している固有値を暗号変換した値（*Response*）をサーバへ返信する．サーバは *Response* を復号し，一致する値を検索してタグを特定する．特定に成功した場合，(3) タグの秘匿情報（固有値など）を更新する *Update* を計算してタグへ送信し，(4) サーバが特定した値を更新する．(5) タグは，受信した *Update* が有効な値であるかを確認して，秘匿情報を更新する．**図 1** に認証プロセスの流れを示す．本稿では，この認証が繰り返し行われると仮定する．

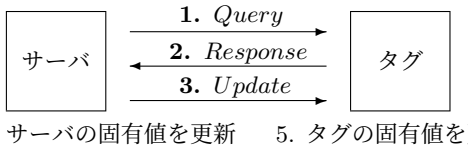

図 1　RFID システムの認証プロセス
Fig. 1　Verification process for RFID system.

2.2 セキュリティ要件

本節では，安全性を保つために必要なセキュリティ要件について述べる．

識別不可能性（IND）6)：*Response* から特定のタグを識別可能である場合，攻撃者はタグの追跡が可能となる．したがって，タグの出力値は匿名でなければならない．

追跡不可能性（BU）6)：耐タンパ性を備えていないタグは，電気解析攻撃などの攻撃によって格納している秘匿情報が攻撃者に漏洩する危険がある．もし，漏洩した秘匿情報を用いて過去の秘匿情報を計算できた場合，攻撃者はタグ所有者の行動履歴を追跡可能となる．追跡不可能性とは，現在のタグの秘匿情報が漏洩しても所有者の過去の行動を追跡できない性質である．

なりすまし攻撃耐性（Spf）7)：攻撃者は正規タグと情報交換することで，タグの秘匿情報を知ることなくタグになりすますことが可能となる．なりすまし攻撃は，以下の攻撃が存在する．RFID システムは，これらのなりすまし攻撃に対して安全でなければならない．

- Passive Attack：攻撃者は，タグとサーバ間の通信を盗聴し，入手した *Response* を正規サーバへ再送信することで正規タグになりすます．

- Active Attack：攻撃者は，正規タグに *Query* を送信し，*Response* を受け取る．そして，受け取った *Response* を正規サーバに送信することで正規タグになりすます．

2.3 同期問題（SP）

ID を用いた相互認証プロトコルにおいて，タグは *Response* をサーバの *Query* ごとに変えるために *ID* を認証ごとに更新する．通常，サーバは，タグの *ID* と同じ *ID* を検索し

© 2011 Information Processing Society of Japan

てタグを識別する．したがって，サーバとタグの ID は認証時においてつねに同期状態でなければならない．同期状態とは，タグの ID とサーバの ID が同じ値である状態を指す．この同期が崩れた場合，サーバは ID の検索に失敗してしまうため，タグの識別が不可能になる．近年，サーバとタグの同期を崩す攻撃が提案されており，本稿では以下の攻撃を考慮する．

Blocking Attack（BA）2)：この攻撃は，サイドチャネルアタックの一種であり，SecureComm 2005 において，Dimitriou らによって提案された．攻撃者は，サーバからタグへ送信された $Update$ を改竄可能であるとする．正規サーバから送信された $Update$ を攻撃者が改竄した場合，タグは受信した $Update$ が不正データであると判断し，秘匿情報の更新を行わない．しかし，サーバ側では $Update$ の送信後に秘匿情報を更新しているため，認証に用いる秘匿情報が非同期の状態となる．

Server Impersonation Attack（SIA）9)：この攻撃は，UBICOMM 2008 において，Song によって提案された．攻撃者は，ある特定のタグ T_i の秘匿情報を入手可能であると仮定する．攻撃者は入手した秘匿情報を格納した偽のサーバを作成し，T_i に $Query$ を送信する．$Query$ を受け取った T_i は，格納している ID を用いて $Response$ を計算して偽サーバに返信する．$Response$ を受け取った偽サーバは，攻撃者が入手した秘匿情報を用いて $Update$ を計算して T_i へ送信する．このとき，偽サーバが T_i から入手した秘匿情報を用いて有効な $Update$ を計算可能ならば，偽サーバは正規のサーバになりすますことができる．偽サーバが送信した $Update$ を受信した T_i は，受信した $Update$ が有効な更新命令であると判断し，秘匿情報の更新を行う．しかし，正規サーバと T_i は認証を行っていないため，正規サーバの秘匿情報は更新されない．したがって，タグとサーバの同期が崩れ，以降の認証はすべて失敗する．

3. 従来のセキュリティ方式

本章では，データ同期を考慮した従来のプロトコルに対する評価を行う．

LOK 方式 5)：タグとサーバに同期化された共有鍵を格納し，ID の同期が崩れた場合は鍵を用いてタグを識別する方式である．しかし，この方式はタグとサーバで ID と鍵を共有しており，サーバがタグへ送信する $Update$ も共有鍵を用いて計算される．したがって，攻撃者はタグの共有鍵を入手することで有効な $Update$ を計算でき，タグの共有鍵を更新することが可能である．SIA 後，ID と共有鍵の同期がともに崩れているためサーバはタグを識別できず，再同期化も行われない．

CC 方式 1)：ハッシュチェイン方式を用いて ID を更新することで，同期が崩れてもサーバ側でタグの ID を計算できる方式である．ID が非同期の場合，サーバは格納している ID のハッシュ値を計算することでタグを識別できる．しかし，この方式は $Update$ を互いに共有している ID を用いて計算する．したがって，攻撃者はタグの ID を入手することで有効な $Update$ を計算することができる．攻撃者は，SIA を 2 回以上行うことでタグとサーバの同期を崩すことができる．このとき，サーバはタグを識別できず，ID の再同期化も行われないので以降の認証はすべて失敗する．

4. 提案方式

本章では，ハッシュ関数，乱数，排他的論理和を用いて 2.2 節で述べたセキュリティ要件を満たし，データ同期を保つプロトコルを提案する．本章における表記法の定義を以下に示す．

Notation

i：値が更新された回数．$i \in (0, 1, 2, 3 \ldots)$.
R：認証開始時に生成される乱数．$R \in (0,1)^m$.
S_i：サーバが持つ秘匿情報．$S_i \in (0,1)^m$.
P_i：前回使用した秘匿情報．$P_i \in (0,1)^m$.
K_i：S_i を暗号変換した値．$K_i \in (0,1)^m$.
N_i：タグが格納している値．初期値は乱数．$N_i \in (0,1)^m$.
$H()$：1 方向性ハッシュ関数．$H\{(0,1)^*\} \rightarrow (0,1)^m$.
$\oplus$：排他的論理和.
$\|$：連結関数.
$\leftarrow$：代入関数.

4.1 提案方式

提案方式は，以下の手順で認証を行う．認証の流れを**図 2** に示す．

(1) **サーバ**：乱数 R を生成し，$Query$ と R をタグへ送信する．
(2) **タグ**：受け取った R と自身が格納している K_i を用いて $N_{i+1} = H(N_i, R \| K_i)$ を計算する．次に，$TK = H(K_i, R \| N_{i+1})$ を計算し，TK と N_{i+1} を $Response$ としてサーバに送信する．送信後，N_i を N_{i+1} に更新する．

© 2011 Information Processing Society of Japan

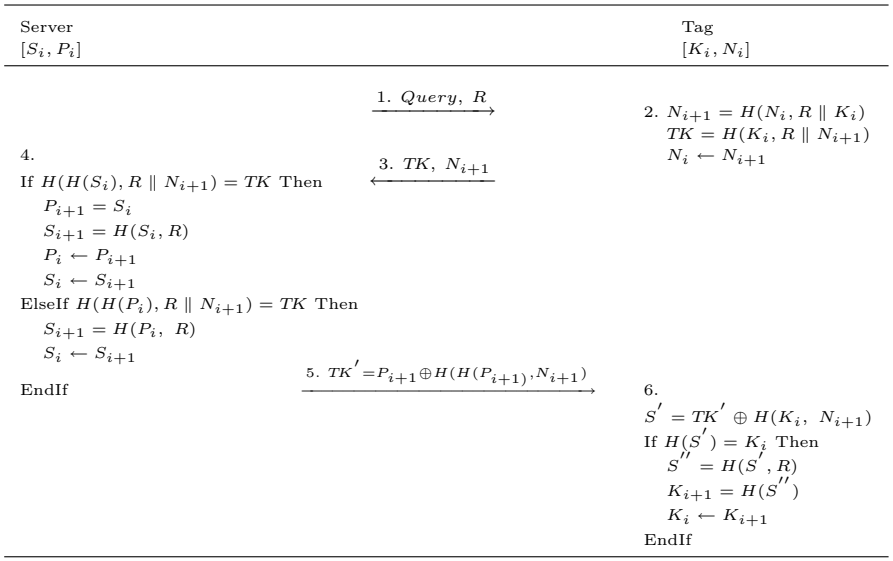

図 2　提案方式の認証プロセス
Fig. 2　Verification process of the proposed scheme.

(3)　**サーバ**：格納しているすべての S_i ごとに $H(H(S_i), R \parallel N_{i+1})$ を計算し，受信した TK と一致する値を計算可能な S_i を検索する．

検索に成功したとき：$P_{i+1} = S_i$, $S_{i+1} = H(S_i, R)$ を計算し，S_i を S_{i+1} に，S_i と対応している P_i を P_{i+1} に更新する．

検索に失敗したとき：S_i を用いてタグを特定できないため，格納している P_i ごとに $H(H(P_i),\ R \parallel N_{i+1})$ を計算して TK と比較する．一致したとき，特定した P_i と対応している S_i を $S_{i+1} = H(P_i, R)$ に更新する．このとき，P_i の更新は行わない S_i と P_i を用いても TK と一致する値を計算できない場合，受信したデータは改竄された可能性が高いと判断し，破棄する．このとき，これ以降の処理は行われない．

(4)　**サーバ**：認証後，$TK' = P_{i+1} \oplus H(H(P_{i+1}),\ N_{i+1})$ を計算してタグへ送信する（P_i が更新されていない場合は，$TK' = P_i \oplus H(H(P_i),\ N_{i+1})$ となる）．この TK' は $Update$ に相当する．

(5)　**タグ**：$S' = TK' \oplus H(K_i,\ N_{i+1})$ を計算する．もし，$H(S') = K_i$ ならば，$S'' = H(S',\ R)$ を計算し，K_i を $K_{i+1} = H(S'')$ に更新する．一致しなかった場合，K_i の更新は行われない．

4.2　セキュリティ評価

本節では，提案方式の安全性について検証を行う．

識別不可能性（IND）：認証開始時，サーバは R を生成してタグへ送信する．この R は認証ごとに生成される異なる乱数である．タグは，サーバからの $Query$ に対して自身の N_i を R, K_i, 1 方向性ハッシュ関数を用いて更新する．よって，N_{i+1} はつねに異なる値となる．また，タグの $Response$ として計算される TK も R, N_{i+1}, 1 方向性ハッシュ関数を用いて計算されるためつねに異なる値となる．さらに，TK と N_{i+1} は 1 方向性ハッシュ関数によって得られた値であるため，$Response$ から K_i や N_i を求めることはできない．したがって，識別不可能性を満たす．

追跡不可能性（BU）：タグが格納する秘匿情報は K_i, N_i である．これらは，1 方向性ハッシュ関数を用いて認証ごとに更新するため，仮に秘匿情報が漏洩しても，過去の秘匿情報を計算できないため，過去の出力値を求められない．したがって，追跡不可能性を満たす．

なりすまし攻撃耐性（Spf）：なりすまし攻撃である Passive Attack と Active Attack に対して，提案方式は安全であることを示す．

- Passive Attack：サーバが生成する R は認証ごとに異なる乱数である．N_i は正規サーバからの $Query$ ごとに R と K_i を用いて N_{i+1} を計算して更新する．TK は R と N_{i+1} によって計算される．よって，タグからサーバへ送信される TK, N_{i+1} は認証ごとに異なる値である．つまり，攻撃者は TK, N_{i+1} を入手しても，次の認証セクションで計算される $Response$ と値が異なるので再利用できない．したがって，提案方式は Passive Attack に対して安全である．
- Active Attack：攻撃者がタグへ送信する乱数を R' とし，正規サーバが持つ乱数を R（$R \neq R'$）とする．攻撃者は R' と $Query$ を正規タグへ送信し，タグは TK, N_{i+1} を計算して攻撃者へ返信する．このとき，$N_{i+1} = H(N_i, R' \parallel K_i)$, $TK = H(K_i,\ R' \parallel N_{i+1})$ である．攻撃者は，受け取った TK, N_{i+1} を正規サーバへ送信する．受け取ったサーバは TK と一致する S_i を検索するために，$H(H(S_i), R \parallel N_{i+1})$ を計算して TK と比較する．しかし，$R \neq R'$ より，$H(H(S_i), R \parallel N_{i+1}) \neq TK$ となり S_i を特定できない．

© 2011 Information Processing Society of Japan

同様に，TK と一致する P_i の特定も失敗する．このとき，サーバは改竄の可能性があると判断して受信データを破棄して認証を行わない．よって，攻撃者はなりすましに失敗する．したがって，提案方式は Active Attack に対して安全である．

4.3 同期問題

タグとサーバの同期を崩す攻撃として，2.3 節で記述した BA と SIA に対して安全であることを示す．

Blocking Attack（BA）：タグが格納している K_i は，サーバが格納している S_i を 1 方向性ハッシュ関数で暗号化した値である．したがって，$H(S_i) = K_i$ が成り立つとき，サーバとタグは同期している状態である．しかし，攻撃者によって $TK^{'}$ を改竄された，またはタグが正常に受信できなかった場合，タグ側の K_i が更新されず $H(S_{i+1}) \neq K_i$ の状態となり非同期の状態になる．この状態のときに通常の認証を行った場合，サーバは前回の認証で用いた S_i を $P_i = S_i$ として格納している．したがって，サーバは P_i を用いることで TK と一致する値を計算できるので，サーバはタグの識別が可能である．また，タグは受信した $TK^{'}$ を用いて，サーバが格納している P_i を得ることができる．この P_i を用いてタグ側の K_i を更新することで再同期化する．

Server Impersonation Attack（SIA）：正規タグ T が格納する秘匿情報は K_i, N_i である．攻撃者は，T が格納する秘匿情報を入手し，それらを格納した偽のサーバ F を作成する．F は乱数 R を生成して $Query$ とともに T へ送信する．$Query$ を受け取った T は格納している秘匿情報を用いて TK, N_{i+1} を計算して F へ返信する．T からの返信を受け取った F は，T の K_i を更新するための $TK^{'}$ を計算する．しかし，F は正規サーバが持つ P_i を格納しておらず，K_i は 1 方向性ハッシュ関数によって求められた値なので K_i から P_i と同じ値を計算できない．したがって，F は有効な $TK^{'}$ を計算できない．T は，$TK^{'}$ から計算した $S^{'}$ を用いて K_i と比較するが一致しないので，タグの K_i を更新しない．したがって，提案方式は SIA に対して安全である．

4.4 従来方式との比較

本節では，提案方式と 3 章で扱った従来方式との安全性の比較を行う．**表 1** では，2.2 節で述べた 4 つのセキュリティ要件と 2.3 節で述べた同期問題に対する安全性を示す．LOK 方式は，タグが格納している ID を暗号変換せずに送信しているため，Active Attack タイプのなりすまし攻撃に対して脆弱である．CC 方式は，タグの ID が更新されなかった場合，タグの出力値が前回の出力と同じ値になる場合が存在するため，識別不可能性と追跡不可能性を満たさない．また，これらの方式は ID や秘密鍵をタグとサーバで共有する方式であるため，SIA に対して脆弱である．しかし，提案方式は 4.3 節で述べたとおり，BA や SIA に対して安全であり，2.2 節で記述したセキュリティ要件もすべて満たしている．

表 1　他の方式との安全性における比較
Table 1　A comparison of security with other schemes.

セキュリティ方式	セキュリティ要件			
	IND	BU	Spf	SP
LOK Protocol [5]	√	√	−	−
CC Protocol [1]	−	−	√	−
Our Protocol	√	√	√	√

（√：要件を満たす，−：要件を満たさない）

5. まとめ

本稿では，従来のセキュリティ要件を満たし，かつ，サーバとタグのデータ同期を保つプロトコルを提案した．提案方式では，サーバに秘密鍵を格納し，タグに秘密鍵を 1 方向性ハッシュ関数で暗号化した値を格納する手法を用いた．仮に攻撃者がタグの秘匿情報を入手しても，1 方向性ハッシュ関数で暗号化されているため，入手したタグの秘匿情報からサーバの秘密鍵を計算できない．攻撃者は正規サーバのなりすましに失敗するので，提案方式はサーバなりすまし攻撃に対して安全であるといえる．また，タグとサーバに同期化した鍵を格納することでデータ同期を保ち，1 方向性ハッシュ関数と乱数を用いることで識別不可能性，追跡不可能性，なりすまし攻撃耐性を満たすことができる．

参考文献

1) Canard, S. and Coisel, I.: Data Synchronization in Privacy-Preserving RFID Authentication Schemes, *RFIDSec. 2008* (2008), available from ⟨http://events.iaik.tugraz.at/RFIDSec08/Papers/⟩.
2) Dimitriou, T.: A Lightweight RFID Protocol to protect against Traceability and Cloning attacks, *SecureComm 2005*, pp.59–66, IEEE Computer Society Press (2005).
3) Juels, A.: RFID security and privacy: A research survey, *IEEE Journal*, Vol.24, pp.381–394 (2006).
4) Kang, J. and Nyang, D.: RFID Authentication Protocol with Strong Resistance against Traceability and Denial of Service Attacks, *ESAS 2005*, LNCS 3813, pp.164–175, Springer-Verlag (2005).

© 2011 Information Processing Society of Japan

5) Lim, J., Oh, H. and Kim, S.: A New Hash-Based RFID Mutual Authentication Protocol Providing Enhanced User Privacy Protection, *ISPEC 2008*, LNCS 4991, pp.278–289 (2008).
6) Ohkubo, M., Suzuki, K. and Kinoshita, S.: Criptographic Approach to *Privacy-Friendly* Tags, *RFID Privacy Workshop 2003* (2003), available from ⟨http://rfidprivacy.media.mit.edu/2003/papers/⟩.
7) Rhee, K., Kwak, J., Kim, S. and Won, D.: Challenge-Response based RFID Authentication Protocol for Distributed Database Environment, *SPC 2005*, LNCS 3450, pp.70–84 (2005).
8) Rieback, M.R., Crispo, B. and Tanenbaum, A.S.: The evolution of RFID security, *PerCom'06, IEEE Pervasive Computing*, Vol.5, pp.62–69 (2005).
9) Song, B.: Server Impersonation Attacks on RFID Protocols, *UBICOMM 2008*, pp.50–55, IEEE Computer Society (2008).
10) Song, B.: RFID Tag Ownership Transfer, *RFIDSec. 2008* (2008), available from ⟨http://events.iaik.tugraz.at/RFIDSec08/Papers/⟩.
11) Song, B. and Mitchell, C.J.: RFID Authentication Protocol for Low-cost Tags, *WiSec 2008, Association for Computing Machinery*, pp.140–147 (2008).
12) Weis, S.A., Sarma, S.E., Rivest, R.L. and Engels, D.W.: Security and Privacy Aspects of Low-Cost Radio-Frequency Identification Systems, *SPC 2003*, LNCS 2802, pp.201–212 (2003).
13) Tan, C.C., Sheng, B. and Li, Q.: Serverless Search and Authentication Protocols for RFID, *PerSec 2007, 5th Annual IEEE International Conference*, pp.235–240 (2007).

(平成 22 年 4 月 14 日受付)
(平成 23 年 6 月 3 日採録)

推　薦　文

従来の ID タグの認証方式のセキュリティ要件である識別不可能性，追跡不可能性，なりすまし攻撃耐性に加え，同期ずれを起こす攻撃に対する耐性さえも持つ新しい方式を提案している．提案方式は，メモリとハッシュ関数，加算（XOR）等，従来のタグの物理要件（タグ基本プロトコルの要件）を満足し，かつ，上記攻撃モデルに対し耐性を持つ．方式の新規性に加え，セキュリティ評価も十分に行われており，推薦論文とするに十分な内容と判断する．　（コンピュータセキュリティ研究会主査　菊池浩明）

丹羽　弘和（正会員）
2009 年公立はこだて未来大学システム情報科学部卒業．2011 年同大学大学院システム情報科学研究科博士前期（修士）課程修了．RFID セキュリティに関する研究に従事．電子情報通信学会会員．

高木　　剛（正会員）
1993 年名古屋大学理学部数学科卒業．1995 年同大学大学院理学研究科修士課程修了．同年 NTT 研究所入社．2001 年 Ph.D.（ダルムシュタット工科大学）．その後，ダルムシュタット工科大学情報科学部准教授，公立はこだて未来大学システム情報科学部教授を経て，2009 年より九州大学数理学研究院教授，現在に至る．暗号および情報セキュリティに関する研究に従事．電子情報通信学会，IACR 各会員．

高橋　　修（フェロー）
1975 年北海道大学大学院工学研究科修士課程修了．同年電電公社（現 NTT）横須賀電気通信研究所入所．コンピュータネットワークの研究・開発・標準化に従事．NTT ドコモを経て 2004 年より公立はこだて未来大学教授．博士（工学）．本会業績賞．本会フェロー．電子情報通信学会，IEEE 各会員．

© 2011 Information Processing Society of Japan