M-010

不正なアプリケーションのなりすましを防止した携帯端末とICカード間の非同期メッセージングの設計

Design of a secure asynchronous messaging framework for mobile phones and smartcards

森　謙作† 寺田雅之† 石井一彦† 本郷節之† 鵜坂智則‡ 越塚登‡ 坂村健‡
Kensaku Mori†, Masayuki Terada†, Kazuhiko Ishii†, Sadayuki Hongo†,
Tomonori Usaka‡, Noboru Koshizuka‡, Ken Sakamura‡

1. はじめに

筆者らは，分散ネットワーク環境上のICカード間の直接通信を実現するフレームワークTENeTを提案している[1]．TENeTはメッセージ配送のためのフレームワークであり，メッセージパッシング方式によりICカードとアプリケーションプログラム（AP）間でメッセージを対称的かつ透過的に配送することができる．

たとえば，携帯電話とそこに挿入されたICカード（USIM）で構成されるシステムにおいて，従来のICカードは携帯電話に搭載されたAPから送付されたメッセージを受理して返答する受動的な処理しかできなかった．しかし，TENeTを適用することでICカードも携帯電話と同様に，ICカードみずからが携帯電話だけでなく他のICカードを宛先としたメッセージを送付できる対称的な構造となる．ICカードが送付したメッセージに対して，APがメッセージを受け取り，宛先を他のICカードに変更するなどの中継処理の必要はない．すなわち，メッセージはAPに対して透過的に配送される．

またメッセージパッシング方式により，2つ以上の複数のICカード間でメッセージを送受することもできる．これらのICカード間で実行される一連の処理において，他のICカードに対してメッセージを送付したICカードは，その返答メッセージを受け取るまで内部処理に空きができるため，携帯電話の利用者（以下，利用者）は送付したメッセージへの結果を待たずに次のメッセージを送付して，処理を行っていないICカードを効率良く利用することも可能である．

この特徴により，利用者はICカードに対してカード内のデータを他のICカードと取引するメッセージを送り，ICカードがその処理を実行中に，さらに次の取引メッセージや処理の中断依頼メッセージを送るなどの柔軟な操作を行うことができる．

メッセージパッシング方式による通信の問題点

しかしメッセージパッシング方式はAPの能動的な操作を可能とする反面，従来のコマンドを受理した順番に送付元へ応答を返すコマンドレスポンス方式と異なり，APの送付メッセージとICカードからの返答メッセージの間の紐付けがないことが問題点として挙げられる．これにより，APは送付メッセージとICカードからの返答メッセージを明確に紐付けることができず，利用者に対して正しい結果を表示できない状況が生じる．

たとえば，利用者がAPを用いてICカードに対してデータの生成依頼メッセージを送付した後に，その返答メッセージを待つことなく，次のデータ生成メッセージを送付する状況を考えてみる．この時APがICカードから1つの返答メッセージを受理しても，返答メッセージがいずれの送付メッセージに対応しているか識別ができず，利用者に対して正しい結果を通知できない可能性がある（図1）．

この問題点への対処方法として，APが送付メッセージの順番と内容を記憶しておき，コマンドレスポンス方式と同様にICカードの返答メッセージを記憶した順番どおりに処理する方法がある．しかしメッセージパッシング方式では，メッセージが複数のICカード間で送受できることから，送付先のICカードやネットワークの状態によっては，メッセージの到着順序が入れ替わる可能性もある．そのため，APは必ずしも送付した順番で返答メッセージを受理できるとは限らない．

そこで，本研究では上記のメッセージパッシング方式の問題を解決した，TENeTにおける送受メッセージの安全な紐付けの方法を提案する．

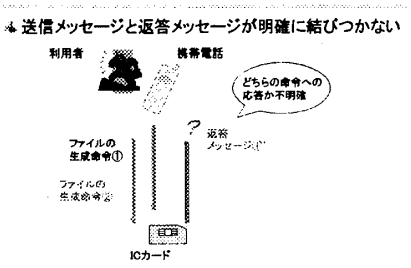

図1 メッセージパッシング方式の問題点

2. TENeTフレームワーク

本章では，TENeTフレームワークの概要を説明する．

2.1. アーキテクチャ

TENeTフレームワークの構成を図2に示す．ここでは簡単のため2端末による構成を示しているが，3者以上においても変わらない．

携帯電話は内部にICカードを挿入しており，携帯電話上にはICカードを利用するための複数のAPが存在する．利用者は用途に応じたAPを用いてICカードにメッセージを送り，ICカード内のデータ処理やICカードを介した他者との通信を行う．ICカードの処理結果はAPに通知され，利用者はAPに通知された内容について，処理完了の確認や処理中の値の了承／否認などの選択を行う．

†株式会社　NTTドコモ　NTT DoCoMo, Inc.
‡東京大学　Tokyo University

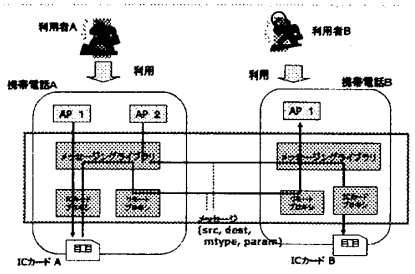

図２　アーキテクチャ

2.2.　メッセージの構成

TENeT アーキテクチャ上で配送されるメッセージは，（**src, dest, mtype, param**）の 4 つ組で構成される．src, dest はそれぞれメッセージの送信元と送信先を示す宛先識別子であり，この宛先識別子には TENeT 上の IC カードが固有に持つ uCode[3]を利用する．ただし，uCode を持たない AP は，IC カードから uCode を拡張した宛先識別子を払い出してもらう．

uCode は 16 バイトの値であり，TENeT では上位 12 バイトを domain (D)，下位 4 バイトを port (P) と呼ぶ．AP に払い出される宛先識別子は，上記 IC カードに固有な D と，IC カードから各 AP に対して一意に払い出される P から構成される (D|P) である．

2.3.　各モジュールの機能

メッセージの配送を行うための TENeT フレームワーク（図 2）の各モジュールの機能を以下に説明する．

メッセージングライブラリは，AP や IC カードから送付されたメッセージを受け取り，プロキシもしくは AP へのディスパッチを行う．dest である AP や IC カードはディスパッチされたメッセージの受理後，mtype で示された操作種別に従って param の内容を用いた処理を行う．

ICC プロキシは，メッセージングライブラリから IC カード宛のメッセージを受理すると，IC カードが受理できるフォーマットに変換して IC カードへ送付する．同様に，IC カードから送付されたメッセージを IC カードへ入力する前のフォーマットに変換してメッセージングライブラリへ送付する．

リモートプロキシは，メッセージングライブラリから送付されたメッセージをネットワーク外へ送付する．メッセージングライブラリは，携帯電話に挿入されている IC カードの uCode を記憶しておき，メッセージの宛先が IC カードの uCode 内の domain と異なる場合は，ネットワーク外の IC カードや AP 宛のメッセージと判断して，リモートプロキシへ送付する．また，メッセージを配送する以外に，外部からの入力メッセージを制限する機能を持つ．たとえば，IC カード内のファイルやフォルダを操作できるメッセージの入力を許可しない，などの制御が可能である．

2.4.　AP に対するインタフェース

TENeT フレームワークは，携帯電話内の AP と IC カード間のメッセージ配送において，AP に対して以下のインタフェースを提供する．

窓口オブジェクト（AP からのメッセージの送付時）

メッセージングライブラリは，AP の初期化時に，AP がメッセージを送付のための窓口となるオブジェクトを生成する．窓口オブジェクトは内部の値として AP の宛先識別子を管理し，メッセージの送付時に宛先識別子を強制的に付与する．AP は窓口オブジェクト以外の方法ではメッセージを送付できない．

メッセージリスナ（AP へのメッセージの通知時）

IC カードや他の AP から AP へメッセージを送付する場合はメッセージリスナを用いる．メッセージリスナは callback 関数と同様に，メッセージを通知してその返答を受け取る役割を持つメソッドである．AP に通知するメッセージに対する返答メッセージの引数としては，ファイルの変更や処理の承諾／拒否の選択など利用者による指定が必要な値のみを与える．

3.　要求条件

本章では，TENeT アーキテクチャ上の AP と IC カード間で送付・返答されるメッセージ間の紐付けを行う方法に対する要求条件を示す．AP が送受されるメッセージを明確に紐付けるために，これらのメッセージを結び付ける紐付けの重複を防止しなければならない．重複の防止を観点において，以下の 2 つの要求条件を抽出した．

AP が意図しない重複の防止

AP 自身の誤りやバグにより，AP と IC カード間でやり取りされるメッセージを紐付けるための情報が重複することを防止するために，送付・返答されるメッセージを確実に 1 対 1 に紐付けられなければならない．また AP からの命令を受け，IC カード間でメッセージをやり取りするなど，複数の AP と IC カード間で行われる一連の処理に用いる複数のメッセージ群も同様に 1 つのグループとして 1 つの情報で紐付けられなければならない．

不正な AP による意図的な重複の防止

利用者が AP を実行して送付したメッセージに対して，不正な AP によりメッセージ間を紐付ける情報が盗み取られてなりすまされ，その結果実行した AP への返答メッセージを横取りされる可能性がある．

従って，不正な AP がメッセージ間を紐付ける情報を，勝手に利用することを防止しなければならない．

4.　提案方法

本章では，要求条件に沿って確立した提案方法について，メッセージの紐付けの方法とメッセージの構成および，メッセージの配送方法について説明する。

4.1.　メッセージの紐付けの方法

2.2 節のメッセージ構成から，従来の TENeT メッセージには送付メッセージと返答メッセージを紐付けるために利用できる値はない．

そこでメッセージを紐付ける方法が必要となるが，紐付けの方法として主に 2 つが考えられる．1 つはメッセージの構成を変更せずに AP が送付メッセージと返答メッセージを対応付ける方法，もう 1 つはメッセージに紐付けのための値を追加する方法である．前者の方法には，1 章で既に述べた AP が送付した順番と受理した順番を対応付ける方法が当てはまるが，メッセージの順序が保証されないため，正確に紐付けられない可能性がある．また，TENeT 上ではメッセージは 3 者以上で送受される場合もある．たとえば，図 3 における左下の IC カードは，一連のメッセージ群を紐付けるために，AP と IC カード間の紐付

けの情報と IC カード間の紐付けの情報を対応付けて記憶するなどの管理が必要となる．後者の方法はメッセージが自身で紐付けの情報を持つため，AP へのメッセージの到着順序に依存せず，AP による管理の手間も削減できる．従って，提案方法では確実に紐付けを行うために後者の方法を採用する．ここで宛先識別子と区別するために，メッセージに含まれる紐付けのための情報をスレッド ID と定義する．

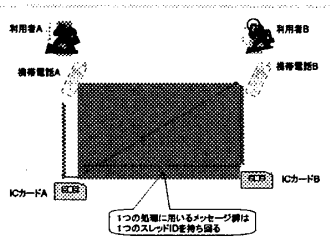

図 3 3者以上のメッセージ送受時のスレッド ID の運用方法

4.2. メッセージの構成

提案方法において，メッセージは 2.2 節のメッセージ構成にスレッド ID を加えた（**src, dest, スレッド ID, mtype, param**）の 5 つ組で構成される．src, dest, mtype, param の内容は従来から変更がない．

本提案では，スレッド ID を（D|P|TID）で構成する．2.2 節で述べたとおり，D は TENeT 内の IC カードが持つ固有の domain であり，P は IC カードが AP 単位に払いだした値である．TID は各 AP において一意である 4 バイトの値である．(D|P)が 16 バイトの値であるため，スレッド ID は 20 バイトの値となる．

4.3. メッセージの配送

メッセージの送付時に，従来の TENeT メッセージに加えて，スレッド ID の付与も行う．スレッド ID は AP 単位で管理が必要な値であるため，AP の宛先識別子と同様に窓口オブジェクトで管理する．スレッド ID は窓口オブジェクト内で生成され，メッセージの送付時に自動的に付与され，メッセージ送信後に値が更新される．

また IC カードから AP へのメッセージの通知時には，メッセージリスナが用いられる．従来の方法では，メッセージリスナは通知する内容を含むメッセージを AP に渡す．AP はメッセージに従った処理を行い，メッセージリスナに対する返答メッセージを送付する．

しかし返答メッセージに含まれるスレッド ID は，2.4 節の機能から AP による新たなスレッド ID が付与される．AP への送付メッセージと返答メッセージのスレッド ID が変更されないように，AP が受理したスレッド ID をそのまま返答メッセージの引数として与えることとすると，AP が意図的にスレッド ID を変更できる可能性が生じる．

そこで提案方法では，AP に対してスレッド ID を渡さずにメッセージリスナにおいて管理する．メッセージリスナは，登録メッセージの到着時に，受信した登録メッセージを引数とする通知オブジェクトを呼び出す．通知オブジェクトは AP に受信メッセージを通知し，AP からの処理結果のメッセージを受け取ると，受信メッセージのスレッド ID をオブジェクト内で返りメッセージのスレッド ID として自動的に付与してメッセージングライブラリへ渡す．

5. 考察

4 章のスレッド ID を用いたメッセージ送受方法について，要求条件と照らし合わせて考察する．また，提案方法による性能への影響について定量的な考察を行う．

5.1. AP が意図しない重複の防止

スレッド ID は 4.2 節のとおり（D|P|TID）で構成されており，D が TENeT 上の IC カード単位で一意であり，P が IC カードへメッセージを送付する AP 単位で一意であることから，(D|P)は TENeT 上の AP を一意とする宛先識別子である．さらに TID がその AP 単位で一意であることから，スレッド ID は TENeT で一意となる．従って 4 章の生成規則に従って動作することで，AP と IC カード間の一往復のメッセージや，AP からのメッセージに対する IC カード間の複数メッセージのやり取りまで，送付メッセージに対してすぐ返答メッセージを返す簡単なメッセージ送受だけでなく，3 つ以上の構成要素間でメッセージがやり取りされる複雑な処理まで含めて，1 つの処理においてスレッド ID が同一であることを保証することができる．従って，D が TENeT 内で一意の値であり，TID が P に対して一意であることから，スレッド ID は TENeT 全体で一意な値となるため，AP が 4 章の内容に従ってメッセージを送付する限り，「AP が意図しない重複の防止」が保証される．

5.2. 不正な AP による意図的な重複の防止

TENeT 上のメッセージ送受において，不正な AP がスレッド ID を重複する機会として，IC カードに対するメッセージの送付時と，IC カードから送付されたメッセージへの返答時の 2 つの機会がある．

それぞれの機会において AP の不正を行おうとした場合，提案方法では，スレッド ID の生成と付与はメッセージライブラリが提供する送信用オブジェクトによる行われるため，不正な操作による不正なスレッド ID の付与はできない．

また，メッセージリスナによる IC カードから AP へのメッセージ通知時には，スレッド ID を引数として与えないため，不正な AP による変更は行えない．

これらの対処方法から，3 章で示した要求条件である「不正な AP による意図的な重複の防止」を実現できる．

5.3. 性能への影響

上記の考察から，提案方式が要求条件を満たしていることが分かった．しかし，スレッド ID の追加による性能への影響が懸念される．そこで，提案方法による性能への影響について，定量的に考察を加える．

4.2 節で示したとおり，スレッド ID は 20 バイトの値である．このスレッド ID の追加によるメッセージ送受時間への影響は，38.4kbps の IC カード R/W を用いて通信する場合，R/W の実効速度は約 25kbps となるため，1 回のメッセージ送信または受信でサイズが 20 バイト増加するにより，6.5msec の遅延が発生する．

また，権利価値を公平に取引可能なプロトコルを TENeT 上に実装[4]し，実際にデータのやり取りを行いスレッド ID の追加による通信時間への影響をみた．スレッド ID を用いない場合の，プロトコル全体の処理時間は約 1.8sec（ネットワークの通信時間は含まない）である．このプロトコルにスレッド ID を適用する場合の処理時間は，IC カードと R/W 間で 8 回の入出力があることから，

上記結果を用いて 52msec の増加となる．これは交換プロトコル全体の処理時間からみて 3%程度の増加である．

6. まとめ

本稿では，メッセージパッシング方式の TENeT に対して，メッセージ間を結び付けるスレッド ID を導入し，スレッド ID を TENeT において一意とする生成方法と，重複を防止した付与方法を提案した．

提案方法において，TENeT 内で一意である識別子を付与されることから AP による意図しない重複を防止でき，また，窓口オブジェクトとメッセージリスナにより紐付けられるメッセージに自動的にスレッド ID が付与されることから不正な AP によるスレッド ID の付与も防止できる．

さらに，実装結果から性能面で約 3%の通信時間の増加に抑えられることを確認した．

特に本稿の提案方式は，TENeT に限らず，Java 環境上の汎用的なメッセージパッシング基盤に適用可能な安全かつ利便性の高い方式である．

参考文献

1) Terada, M., Mori, K., Ishii, K., Hongo, S., Usaka, T., Koshizuka, N., and Sakamura, K.: TENeT: A Framework for Distributed Smartcards, In Proceedings of the Second International Conference on Security in Pervasive Computing (SPC 2005), 2005.
2) ISO/IEC Integrated circuit(s) cards with contacts – Part4: Interindustry commands for interchange, ISO/IEC 7816-4: 1995(E).
3) 坂村健，越塚登：「ユビキタス ID センターの取り組み」，月刊バーコード, vol. 16, no. 5, 日本工業出版, 2003.
4) 森謙作，寺田雅之，石井一彦，本郷節之：「安全な電子価値交換プロトコルの IC カード実装」，第 27 回 CSEC 研究会，2004.