

## モニタリングによる WAP の通信手順の 解析方法に関する一検討

石川 彰夫

井戸上 彰

加藤 聰彦

KDD 研究所

### 1. はじめに

モバイル通信の普及に伴い、ワイヤレスデータ通信に対する専用プロトコルである WAP (Wireless Application Protocol) が注目されている。筆者らは、WAP を対象として、通信手順の詳細解析をおこなうプロトコル・アナライザの開発を進めている[1]。このアナライザはネットワークを流れる WAP の PDU (Protocol Data Unit) を取り込み、通信を行うシステム (クライアントとサーバ) 内の WAP モジュールの動作を推定することを基本とする。WAP は、要求と応答の PDU を確実に転送する WTP (Wireless Transaction Protocol) [2]、クライアントがサーバにアクセスして情報を検索するセッションを管理する WSP (Wireless Session Protocol) [3]など、複数のプロトコルから構成される。本稿では再送などの複雑な手順を持つ WTP に着目し、本アナライザにおける解析方法について述べる。

### 2. WTP の概要

クライアント (移動端末) とサーバ間で交換される、要求とそれに対する応答の組み合わせは「トランザクション」と呼ばれる。WTP では、トランザクションを開始する側を「イニシエータ」、他方を「レスポнда」と呼ぶ。トランザクションにはイニシエータが付与する識別子 (トランザクション識別子。以下、TID) が与えられる。トランザクションは、アドレス4つ組 (発信元・宛先の IP アドレス・ポート番号) と TID の組み合わせにより特定される。

トランザクションには、要求とそれに対する応答を転送できるクラス 2、受信確認付きで要求を転送するクラス 1、単に要求のみを転送するクラス 0 の 3 種類がある。クラス 2 で使用する PDU を表 1 に示す。また WTP の主機能を以下に示す。

#### 2.1. 再送

送信した PDU に対する Ack の応答 (あるいは Result-PDU の返信) が一定時間以上無い場合、この PDU の伝送は失敗したと判断し、同じ PDU を送信する。クラス 2 では Invoke と Result の両 PDU が再送の対象となる。

#### 2.2. TID 検査

イニシエータがレスポндаに同じ Invoke-PDU を

重複して送信しないように、Invoke-PDU の TID が重複していないかを検査する機能である。レスポндаは過去受信した TID よりも小さい TID を持つ Invoke-PDU を受信すると、TID の検査を行う旨を通知する Ack-PDU (Tve 付き Ack) をイニシエータに返す。イニシエータは指示された TID に対応するトランザクションを実行中の場合、TID が正当である旨を通知する Ack-PDU (TIDok 付き Ack) を返す。そうでない場合は Abort-PDU を送信しトランザクションを中止する。

#### 2.3. ユーザ Ack

Invoke-PDU の受信をユーザレベルで確認するための機能である。この機能を使用する場合は、イニシエータが Invoke-PDU を送信する際に、フラグ (U/P フラグ) を立てる。

表1 WTP の PDU 種別

名称	機能
Invoke	ユーザからの要求を転送する PDU。イニシエータがレスポндаへこの PDU を送信することにより、トランザクションが開始される。本 PDU を送信する時点で、イニシエータは、TID の値を1ずつ進める。
Result	ユーザからの応答を転送する PDU。レスポндаからイニシエータへ返送される。
保留 Ack	レスポндаからの Result-PDU の送信が遅れた場合に、イニシエータからの Invoke-PDU の不必要な再送を避けるため、レスポндаが正しく Invoke-PDU を受信したことを通知するための PDU。
ラスト Ack	イニシエータから送出される Result-PDU 応答用の PDU。
Abort	トランザクションを中止するための PDU。ユーザの要求によるユーザ Abort と、WTP のエラーにより送信されるプロバイダ Abort の 2 種類がある。

さらに、WTP のクラス 2 を処理するために定義されている内部状態を表 2 に示す。

表2 WTP の内部状態

イニシエータ		レスポнда	
名称	状態	名称	状態
NULL	トランザクションが実行中でない	LISTEN	Invoke-PDU を待つ
RESULT WAIT	Invoke-PDU の送信後、Result-PDU を待つ	TIDOK WAIT	TID 検査の開始後、TIDok フラグ付き Ack-PDU を待つ
RESULT RESP WAIT	Result-PDU の受信後、ユーザからの応答を待つ	INVOKE RESP WAIT	Invoke-PDU の受信後、ユーザからの応答を待つ
WAIT TIMEOUT	Result に対する Ack-PDU を送信後一定時間待つ	RESULT WAIT	ユーザからの Result 送信要求を待つ
		RESULT RESP WAIT	Result-PDU の応答を待つ

### 3. WTP の通信手順の解析方法

#### 3.1. 設計方針

本アナライザでは、ネットワークを転送される PDU を監視し、PDU の送信および受信をイベントとして、イニシエータおよびレスポンドの内部状態の遷移を推定する。このために、アドレスや状態を保持するトランザクション管理テーブルを、発生したトランザクションごとに用意する。

PDU 送信イベントに伴う内部状態の推定においては、そのイベントから、対応する内部的な動作を推定する必要がある。しかし、1つの PDU の受信に対する遷移が複数あり得る場合は、次に遷移する状態を一意に推定することができない。そこで、そのような場合は、遷移後に出力される PDU を調べることにより区別を行う。すなわち、その PDU を受信した後の送信イベントを先読みすることにより、内部状態遷移の候補を絞ることとする。アナライザの起動直後や PDU の取りこぼしが発生した後は、過去の状態遷移の過程が不明であるため、次に遷移する状態を推定できない場合がある。このため、仮の状態として UNKNOWN を定義する。遷移先が決定できない場合や、イベント PDU の取りこぼしなどにより次に遷移すべき状態が存在しない場合は、次の状態を UNKNOWN とする。

上記を考慮して、アナライザ専用の WTP 状態遷移を定義する。

#### 3.2. トランザクション管理テーブル

トランザクション管理テーブルの構造を表 3 に示す。トランザクション管理テーブルは、インデックス部分とデータ部分からなる。インデックス部分は TID およびアドレス 4 つ組を指し、トランザクションを特定するために使用される。それ以外の、状態などの各種情報がデータ部分であり、状態遷移推定に必要な情報を記録するために使用される。

表3 トランザクション管理テーブルの構造

項目	内容
TID	トランザクション識別子
発信元IP	発信元のIPアドレス
宛先IP	宛先のIPアドレス
発信元ポート	発信元のポート番号
宛先ポート	宛先のポート番号
状態	現在の内部状態の推定値
プロバイダタイプ	イニシエータとレスポンドの区別
クラス	トランザクションのクラス
前回のTID	イニシエータが前回使用したTID。イニシエータがInvoke-PDUを送信する度に、現在使用しているTIDに更新する。
再送回数	再送の回数。再送がある度に1ずつ増える。
最大再送回数	可能な再送回数の上限の推定値。再送回数がこの値より大のとき、値を更新する。
保留Ack	保留Ack-PDUを送信した。
TIDok	TIDokフラグを立てたAck-PDUを送信した。
ユーザAck	ユーザAck機能を実行中。U/P=TRUEのInvoke-PDUを送受信するとTRUEになり、U/P=FALSEのInvoke-PDUを送受信するとFALSEになる。

#### 3.3. アナライザのための内部状態遷移

本アナライザで用いるクラス 2 のトランザクション用の内部状態遷移図を図 1 に示す。前述のように、仕様書に基づく状態遷移とは異なり、先読み結果に基づく分岐や、状態 UNKNOWN の追加がなされている。

例として、モニタリング開始直後に、Invoke-PDU の送受信があった場合を考える。図 1(a) から、TID の値が前回の TID よりも大きい場合は、イニシエータの内部状態が RESULT WAIT に遷移したことを確定する。一方、レスポンドの内部状態は一意には確定できないため、PDU の先読みを行う。その結果、トランザクション ID 検査用 Ack-PDU が確認されなければ、レスポンドの内部状態が INVOKE RESP WAIT に遷移したと確定する。

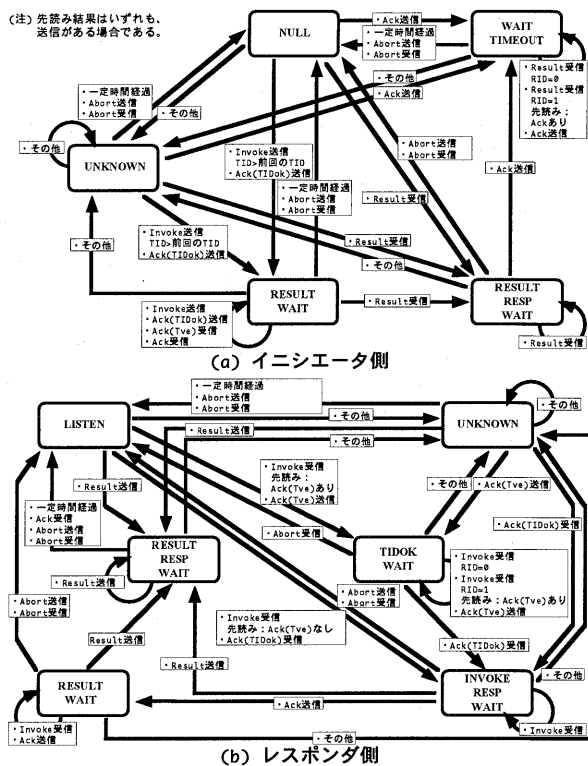


図1 WTP の内部状態遷移図

#### 4. おわりに

本稿では、WAP に従った通信で転送される PDU を収集し、WAP の仕様に基づいて通信システムの WTP の内部的処理の内容を推定し、通信手順を解析するプロトコル解析アナライザの詳細設計について述べた。最後に、日頃ご指導頂く KDD 研究所秋葉所長に感謝する。

#### 参考文献

- [1] 石川他: “WAP 用プロトコル解析ツールの設計”, 第 61 回情報大会, 6H-06 (2000.10).
- [2] WAP Forum, “Wireless Transaction Protocol Specification,” June 1999.
- [3] WAP Forum, “Wireless Session Protocol Specification,” Nov. 1999.