WAP 用プロトコル解析ツールの設計

6H-06

石川 彰夫　　前島 治　　井戸上 彰　　加藤 聰彦
KDD 研究所

1. はじめに

無線ネットワークを介して効率的にデータ通信サービスを提供するために、WAP (Wireless Application Protocol)の標準化が進められており、それに基づくインターネット型サービスが開始されている。その普及に伴い、ネットワークやサーバの輻輳等の原因により各種の障害が発生することが予想される。このような障害を検出しその原因を解析するためには、ネットワークを流れるデータを収集し、WAP に従ってその通信手順を解析する必要がある。そこで筆者らは、ネットワークを流れるデータに基づいて、通信を行うシステムの内部的処理を推定し、それに基づいて通信手順の詳細を解析する[1]WAP 用のプロトコル解析ツールを開発している。本稿ではその設計の概要について述べる。

2. 設計方針

（1）本ツールは、図 1 に示すように、WAP に従った通信が行われているネットワークセグメントに挿入され、ネットワークを流れるデータ(PDU: Protocol Data Unit)を収集し、WAP 通信のクライアントとサーバにおけるプロトコル手順の解析を行う。

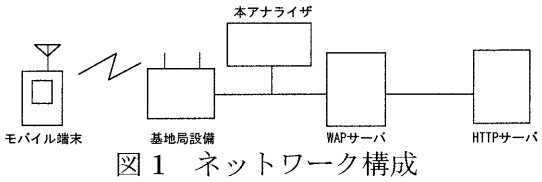


図 1　ネットワーク構成

（2）プロトコル手順の解析においては、特定のクライアント・サーバ間の通信に着目し、そのクライアントとサーバの内部における WAP プログラムの動作を推定する。その結果、その通信のシーケンス図を表示し、データと応答の対応、再送データとオリジナルデータとの対応を明示する。またクライアント・サーバにおける WAP プログラムに誤りがあるかどうかを検知しその誤りを明示する。

（3）本ツールは、オンラインで PDU を収集しファイルに保存し、オフラインで、特定のクライアントとサーバの組（一方のみの指定でも可）に対して解析を行う。

（4）本アナライザは、WAP のプロトコルの内、UDP 上で動作する WTP (Wireless Transaction Protocol)[2]、WSP (Wireless Session Protocol)[3]を対象とする。また、ネットワークインタフェースとしては、10Mbps および 100Mbps の Ethernet を対象とする。

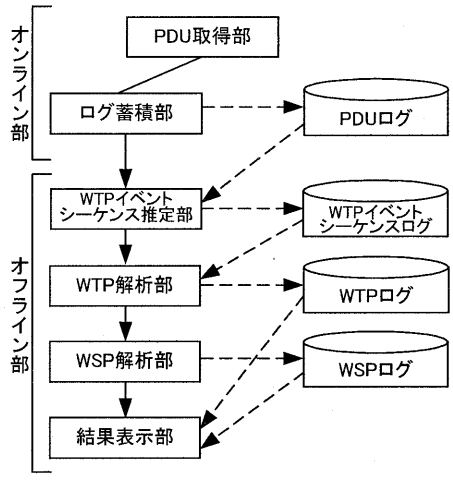
3. ソフトウェア構成

3.1. 概要

図 2 に本ツールのソフトウェア構成を示す。前述のように本アナライザは、オンライン部とオフライン部から構成され、オンライン部ではネットワークを流れる PDU を取得し、ログ蓄積部で検出時刻とともに PDU ログに蓄積する。

オフライン部では、まず PDU ログの情報をもとに、クライアントまたはサーバにおける WTP のイベントの識別を行う。ここでは、単一の WTP PDU に対してクライアント側およびサーバ側のいずれが、送信・受信イベントを起こしたかを識別し、WTP イベントログに記録する。例えば、Invoke PDU が検出されると Invoke 送信イベントと Invoke 受信イベントを識別し、それぞれの推定発生時刻および PDU ログ中のデータへのポインタとあわせて、ログに記録する。なお、イベントの推定発生時刻とは、PDU の検出時刻（その PDU の最後のビットが転送された時刻）から、送信側において、その PDU の最初のビットが送信された時刻、受信側において最後のビットが受信された時刻を、それぞれ推定したものである。

図 2　プログラムの構成



次に、WTP 解析部では、WTP イベントシーケンスログから、1 つ 1 つイベントを取り出して、その処理を行う。この処理においては、WTP PDU に対応する送受信 IP アドレスとポートの組（アドレス 4 つ組）およびトランザクション ID の値などから、他の PDU との関連付けを行い、WTP の状態遷移や内部パラメータ更新の確認、分割さ

"Design of WAP Protocol Analyzer" by Akio Ishikawa, Osamu Maeshima, Akira Idoue and Toshihiko Kato
KDD R&D Laboratories

れたデータのリアセンブリングなどを行う。これらの結果を WTP ログに記録する。さらに、リアセンブリングの結果、WSP の PDU が完成すると、その(WSP)イベントを WSP 解析部に渡す。

WSP 解析部に渡されたイベントは、WTP イベントの種別により、送受信の区別がつけられている。WSP 解析部では、プロトコルに従い、セッションの状態管理、Method の状態管理、Push の状態管理などを行い、それらの結果を WSP ログに記録する。

結果表示部は、オペレータの表示要求に応じて、必要な情報を WTP/WSP ログから検索して表示する。

3.2. 手順解析の詳細

WTP および WSP は、WTP におけるクラス 2 トランザクションが、セッションの確立（S-Connect）や要求（S－MethodInvoke）や応答（S－MethodResult）などと対応しているなどの、関連性を有する。このため、WTP 解析部と WSP 解析部は、図 3 (a)のような統一的なデータ構造を用いて、プロトコルの手順解析を行う。なお、図 3 のデータ構造はクライアント側とサーバ側の双方に対して作られるとする。また、過去に検出していないアドレス 4 つ組の WTP PDU を受信した場合、WTP PDU に含まれる WSP PDU を解析しない限り、クライアント側とサーバ側のどちらに対応させるべきかが判定できない。このため、初期のトランザクションを管理するためのデータ構造として、図 3 (b)に示すデータ構造を用いる。

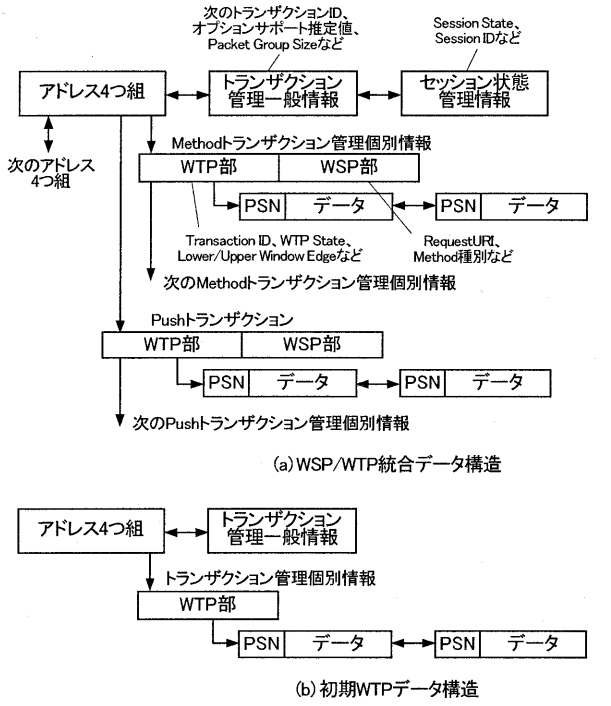
これらを用いた具体的な手順は以下のようになる。

（1）1 つの WTP PDU を検出すると、そのアドレス 4 つ組を用いて、クライアント側およびサーバ側のデータ構造を検索する。対応するエントリが存在しない場合、例えば、セッションの先頭の WSP PDU (Connect PDU) を含む WTP InvokePDU 送信イベントの場合、図 3 (b)に示す構造によりトランザクションの管理を行う。この構造では、アドレス 4 つ組から受信したトランザクションを管理するトランザクション管理個別情報をポイントする。ここにはトランザクション ID などの WTP に関する情報が存在し、Invoke PDU が分割された場合は、完全な WSP PDU が組み立てられるまで、PSN (Packet Sequence Number)とデータをリストでデータ要素が存在する。またあわせて、アドレス 4 つ組からトランザクション管理の一般情報を生成し、そこに次に期待するトランザクション ID などの値を書き込むとともに、WTP ログに必要な情報を記録する。

（2）次に、WSP PDU が完成すると（現在の例では 1 つの WTP PDU が完全な WSP PDU を含む）、その PDU が WSP 解析部に渡される。その時点で WSP PDU の種別が判定できるため、その PDU を送受信したシステムのいずれがクライアントかサーバかが判別される。その結果初期 WTP データ構造にあった情報が、対応する WSP/WTP 統合データ構造に移され、解析された上で、セッション状態管理状態やトランザクション管理個別情報に必要な情報が記録される。現在の例では、Connect PDU の送信イベントであるため、クライアント側のデータ構造に、このアドレス 4 つ組が登録され、トランザクションの情報は Method トランザクション管理個別情報として保持される。その後、Connect PDU のパラメータが解析され、Proposed Capability や Headers などの情報がセッション状態管理情報に保持される。さらに必要な情報を WSP ログに記録する。

（3）WSP 解析部の処理が終了すると、トランザクションのデータが破棄される。

図 3　手順解析用データ構造



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、それに基づいて通信システムの WTP・WSP の内部的処理を推定し、通信手順の詳細を解析するプロトコル解析ツールの設計について述べた。現在その実装を進めているところである。最後に、日頃ご指導頂く KDD 研究所秋場所長に感謝します。

参考文献

[1] Toshihiko Kato, et.al., “Design of Protocol Monitor Emulating Behaviors of TCP/IP Protocols,” IWTCS’97, pp.416-431, Sept. 1997.

[2] WAP Forum, “Wireless Transaction Protocol Specification,” June 1999.

[3] WAP Forum, “Wireless Session Protocol Specification,” Nov. 1999.