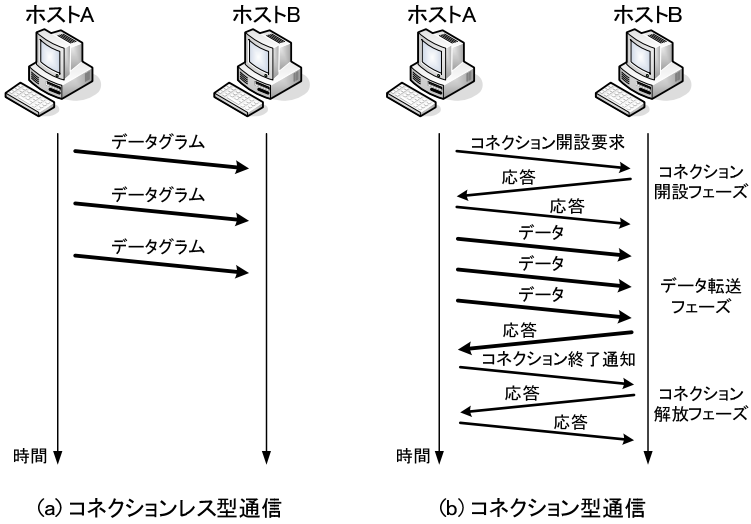
■3 群（コンピュータネットワーク）－4 編（トランスポートサービス）

4 章 UDP (User Datagram Protocol)

（執筆者：相原玲二）［2013 年 6 月 受領］

■概要■

User Datagram Protocol（以下，UDP）は，TCP/IP 通信方式で使用されるトランスポート層プロトコルの一つである．UDP は，ネットワーク層プロトコルとして IP（IPv4 または IPv6）が利用されることを前提として設計されている．UDP はトランスポート層の位置付けであるが，IP の機能を概ねそのまま利用することを目的としており，その仕様は非常に単純である．代表的なトランスポート層プロトコルである TCP はコネクション型として設計されているが，UDP はコネクションレス型（**図 4･1**(a)）であり，パケット紛失やデータ誤りに対する再送制御などの機能を含んでいない．また，ネットワークを経由することでパケットの到着順序が送信順序と異なる場合もあるが，それを補正する機能も含んでいない．一方，コネクション型（図 4･1(b)）では通信の開始や終了のために一定の手順と処理が必要であるが，コネクションレス型の UDP ではその必要がないため，用途によっては処理が簡単である利点を活かすことができる．例えば，DNS，NTP，SNMP などのアプリケーション層プロトコルが UDP を利用している．また，音声や映像のリアルタイム伝送でも，UDP の利点を活かし利用されることが多い．更に，一つのパケットをルータで複製するなどして複数の相手に同時に伝送する IP マルチキャスト通信では UDP が使用されている．



(a) コネクションレス型通信　　(b) コネクション型通信

図 4・1　コネクションレス型通信とコネクション型通信

■3群 － 4編 － 4章

4-1 UDP通信方式

(執筆者：相原玲二)[2013年6月 受領]

UDPにおける送信情報の単位をデータグラム（Datagram）と呼ぶ．これはIPと同様である．UDPとIPを明確に区別したい場合，それぞれUDPデータグラム，IPデータグラムと呼ぶこともある．現在では送信情報の単位をパケットと呼ぶことも多いため，UDPパケット，IPパケットと呼ばれることもある．

UDPはホスト（端末）同士で接続（コネクション）確立の手順を踏まない通信（図4･1(a)）を行う．そのため，宛先アドレスや送信元アドレスなど，そのデータグラムを送り届けるために必要な情報をすべてのデータグラムに含めておく．送信端末から送り出されたデータグラムは，中継装置（ルータ）がデータグラム中のヘッダ部分に含まれる宛先アドレスなどに従って適切な経路選択を行うことで宛先端末に届けられる．送信端末から宛先端末に向けて複数のデータグラムが連続して送信されたとしても，途中の中継装置の経路選択は動的に変化することも考えられるため,データグラムが送信順序どおり宛先端末に届くとは限らない．また，途中の通信回線の混雑状況や中継装置の状態により，データグラムが宛先端末に届かず，破棄されてしまうこともあり得る．このような通信方式をコネクションレス型通信またはデータグラム通信と呼ぶ.

■3群 － 4編 － 4章

4-2 UDP利用例

（執筆者：相原玲二）［2013年6月 受領］

コネクション型では接続の確立や終了のために一定の手順と処理が必要であるが，コネクションレス型のUDPではその必要がないため，用途によっては処理が簡単である利点を活かすことができる．ここでは，アプリケーション層プロトコルDNS（Domain Name System）のUDP利用例を紹介する．DNSはインターネット上のドメイン名形式（example.comなど）とIPアドレス（192.41.192.129など）を対応づけ，特定のホストをドメイン名形式で指定すると対応するIPアドレスを検索する機能（**図4･2**）を提供する．一つのホストをドメイン名形式で質問する場合，多くの場合，問合せの情報は一つのデータグラムに収まると期待されること，利用者（クライアント）の増加により問合せは多数発生することなどから，コネクション確立の不要なUDPが適している．また，DNSにおいては，障害などのためサーバが一時的に利用できない事態を想定し，問合せに対する回答が得られない場合，同じ情報をもつ別サーバへ問い合わせる機構（図4･2の⑥，⑦）が組み込まれている．そのため，UDPでの通信は途中の回線混雑などの影響でデータグラムが損失する可能性があるものの，別サーバへの問合せなどにより目的は達成できると考えられる．

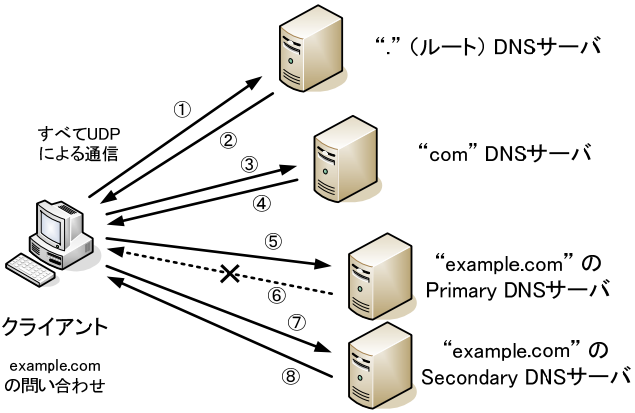


図4･2 DNSにおけるUDPの利用

電話やテレビ会議では，音声や映像を可能な限り遅延なく伝送（リアルタイム伝送）することが求められる．TCPではパケット損失に対して該当パケットの再送で対応するが，再送の必要性の判断及び再送パケットが宛先に届くには一定時間が必要となり，リアルタイム伝送には向いていない．そのため，音声や映像のリアルタイム伝送にはUDPを利用する場合が多い．リアルタイム伝送による音声や映像では，ある程度のUDPデータグラム損失は許容できる場合が多い．また，あらかじめ冗長情報を付加して送信し，データグラムが損失した場合，受信側で補償するFEC（Forward Error Correction）方式（**図4･3**）を用いることもできる．FEC方式で用いられる冗長情報の生成方法としてはパリティ符号やReed-Solomon符号など

があり，付加される冗長情報量と損失に対する耐性は符号化方式により異なる．通常，トランスポートプロトコルとして UDP を用い，アプリケーションプロトコルなどにおいて FEC 機能が用いられる場合が多い．

なお, リアルタイム伝送のためのトランスポートプロトコルとして RTP（Real-time Transport Protocol）が知られている．RTP パケットに付加された時刻情報を基に，宛先ホストではパケットの時間的な関係を把握したうえでデータ処理ができる．これにより受信側で期待する時間以内に到着したパケットだけを利用することが可能となる．通常，RTP は UDP データグラムのデータ部として伝送されている．

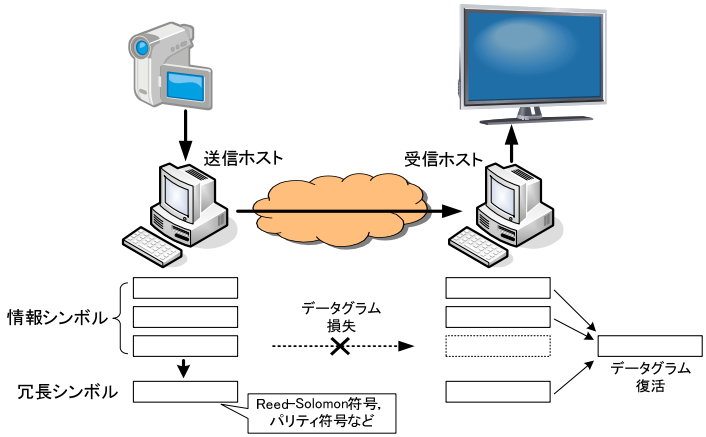


図 4・3　音声や映像のリアルタイム伝送における UDP の利用

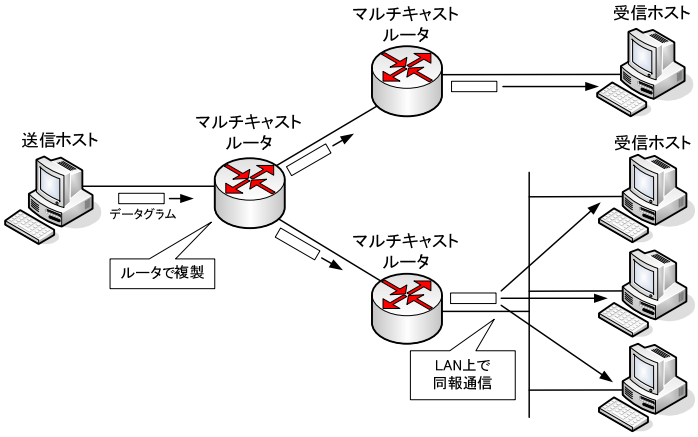


図 4・4　IP マルチキャスト通信における UDP

TCP などのコネクション型通信では，通信を行う 2 端末間においてコネクションの開設を行い，その端末間の通信状況を確認しながら情報伝送を行う．したがって，一つの IP データグラムをルータで複製するなどして，複数の宛先に同じデータグラムを同時に伝送する IP マルチキャスト通信（**図 4･4**）では TCP を使用することはできないため UDP が使用されている．

その他，TCP が提供する再送アルゴリズムやフロー制御アルゴリズムが適さないなどの理由からアプリケーションが独自に再送手順またはフロー制御方式をもつ場合，トランスポート層プロトコルとしては UDP が使用されることがある．

■3群 － 4編 － 4章

4-3 UDP フォーマット仕様

（執筆者：相原玲二）［2013年6月 受領］

UDPにおいて用いられているUDPヘッダのフォーマットを**図4･5**に示す．UDPヘッダはIPヘッダの直後に送信される．なお，IPヘッダ内にはトランスポートプロトコルの種類を示すプロトコルフィールドがあるが，UDPのプロトコル番号は17（10進数表記）と定義されている．

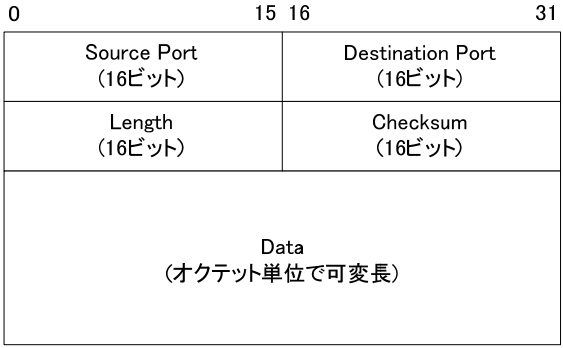


図4･5 UDPヘッダフォーマット

送信元ポート（Source Port）フィールドと宛先ポート（Destination Port）フィールドはUDP処理の種類を区別するために用いられる．送信元ポートはオプションで，使用しない場合は0が指定されることになっている．送信元ポート番号に0以外が指定された場合，送信側の処理プロセスを区別するための番号として使用される．宛先ポート番号は必須で，宛先ホスト上での受信処理プロセスを区別するために使用される．

長さ（Length）フィールドはUDPデータグラムの長さをオクテット単位で示すもので，UDPヘッダ（8オクテット）とデータの合計である．UDPヘッダは必ず存在するため，長さの最小値は8である．

チェックサム（Checksum）フィールドは，伝送されたUDPデータグラムの情報に誤りが含まれているか否かを確認するために使用される．チェックサムは16ビット単位で1の補数和を計算することにより求められる．チェックサム計算の対象は，IPヘッダに含まれるIPアドレスなどから作成される擬似ヘッダ（IPv4とIPv6では形式が異なる），UDPヘッダ及びデータである．なお，データはオクテット単位で指定できるため，奇数オクテットの場合は1オクテット（8ビット）の0が追加（パディング）されたものとして計算することになっている．ただし，擬似ヘッダ及びデータが奇数オクテットの場合のパディングはチェックサムの計算に使用されるだけであり，データグラムとしては送信されない．

IPv4で使用されるUDPヘッダにおけるチェックサムはオプションである．チェックサムを使用しない場合，チェックサムフィールドには0が指定されて送信される．チェックサムを使用する場合，たまたま計算結果が0となることもあるが，その場合はチェックサムフィールドの全ビットを1とする．1の補数において全ビットが1である場合も0を示すため，

チェックサムがオプションであるか否かを上記の方法で区別することが可能である．なお，IPv6 では UDP チェックサムが必須であると定義されている．

チェックサムの計算対象として擬似ヘッダが含まれている．IPv4 及び IPv6 における擬似ヘッダのフォーマットを**図 4･6** 及び**図 4･7** に示す．

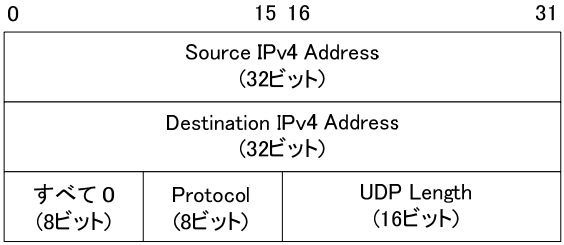


図 4・6 IPv4 擬似ヘッダフォーマット

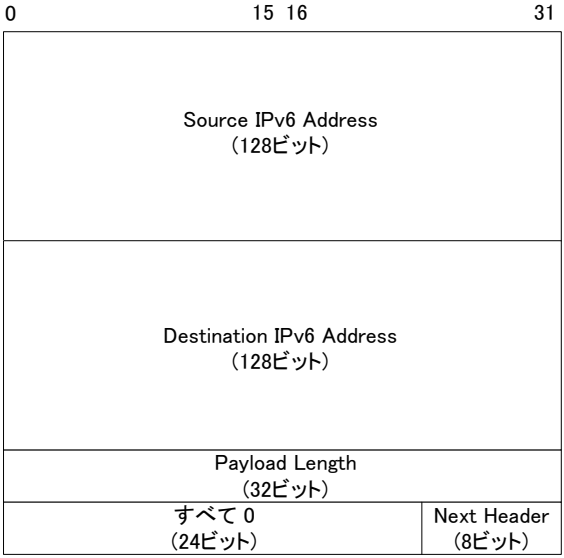


図 4・7 IPv6 擬似ヘッダフォーマット

送信元アドレス（Source Address）及び宛先アドレス（Destination Address）にはいずれも IP アドレスが含まれる．UDP 長（UDP Length）（IPv4）及び上位層パケット長（Upper-Layer Packet Length）（IPv6）は UDP データグラムの長さ（UDP ヘッダ及びデータ）である．また，プロトコル（Protocol）（IPv4）及び次ヘッダ（Next Header）（IPv6）にはプロトコル番号（17）が含まれる．

チェックサムの対象として擬似ヘッダが含まれているのは，ネットワーク層以下（IP 層及びリンク層）において誤りの検出ができず，本来受け取るはずのないデータグラムを受け

取ってしまった場合に対処するためである．しかし，IPv4 においては IP ヘッダチェックサムによる確認があるため，該当することは稀である．IPv4 で UDP チェックサムがオプションとされている理由の一つであろう．一方，IPv6 においては IP ヘッダチェックサムの機能は削除されたため，UDP チェックサムが必須条件に変更されたものと思われる．

■3群 － 4編 － 4章

4-4 UDPの拡張

(執筆者：相原玲二)[2013年6月 受領]

電話，テレビ会議，リアルタイムオンラインゲームなど遅延に敏感で実時間性の求められるアプリケーションのトランスポート層プロトコルとしてはTCPよりもUDPが用いられる．インターネットの広帯域化・低遅延化にともない，これらリアルタイム伝送を要求するアプリケーションの利用は増加する．もともとUDPはDNS情報などの小さなトラフィックを無駄なくやり取りすることを想定していたことから輻輳制御の機構が含まれていない．TCPなどフロー制御を行うトランスポート層プロトコルの通信と混在し，輻輳が発生した場合，公平性の点で問題が発生する．音声や映像などのリアルタイムアプリケーションがUDPをそのまま使用して，通信回線の帯域限界まで通信を行った場合，同じ回線を使用するTCPを用いたアプリケーションの通信ができないという事態が懸念される．そこで，UDPに輻輳制御を追加したトランスポート層プロトコルとしてDCCP（Datagram Congestion Control Protocol）が提案されている．DCCPは輻輳制御機構のみを追加し，パケットが損失した場合の再送制御は含まれていない．DCCPでは複数の輻輳制御アルゴリズムを使用することができ，通信開始時のネゴシエーションにより使用するアルゴリズムを決定する．TCPと類似のウィンドウフロー制御による輻輳制御，TFRC（TCP Friendly Rate Control）と呼ばれるアルゴリズムによる輻輳制御などが選択できる．TFRCは，同じネットワーク環境においてTCPと同じ平均帯域を得られるようなレート制御を行う．これらを用いることにより回線を共有するTCPフローに大きな影響を与えることなく通信を行うことができるようになる．

また，UDPに対しセキュリティ機能を拡張したDTLS（Datagram Transport Layer Security）が提案されている．TCPのセキュリティ機能拡張であるTLS（Transport Layer Security）と類似の機能をデータグラムの特徴を考慮して設計されている．IPデータグラムに対するセキュリティ機能としてIPsecが提案されているが，IPsecは利用できる用途が限られるなどの制約があるため，トランスポート層におけるデータグラムセキュリティ機能としてDTLSが提案されている．

■参考文献

1) RFC-768 User Datagram Protocol, Aug. 1980.
2) RFC-791 Internet Protocol, Sep. 1981. (IPv4)
3) RFC-793 Transmission Control Protocol, Sep. 1981.
4) RFC-1071 Computing the Internet Checksum, Sep. 1988.
5) RFC-2460 Internet Protocol, Version 6 (IPv6) Specification, Dec. 1998.
6) RFC-3550 RTP: A Transport Protocol for Real-Time Applications, Jul. 2003.
7) RFC-4340 Datagram Congestion Control Protocol (DCCP), Mar. 2006.
8) RFC-3448 TCP Friendly Rate Control (TFRC): Protocol Specification, Jan. 2003.
9) RFC-4347 Datagram Transport Layer Security, Apr. 2006.
10) RFC-5246 The Transport Layer Security (TLS) Protocol Version 1.2, Aug. 2008.