

IP マルチキャストグループを制御するプロトコル

篠田 晃

poripori@slab.ntt.co.jp

NTTソフトウェア研究所

概要

IP(Internet Protocol)において同時に複数人と通信する方法としてIPマルチキャストがある。IPマルチキャストでは複数対向への通信の範囲としてグループを形成する。MBone などではオーディオ／ビデオのデータによる放送型の通信形態が多いが、複数マシンへのプログラムやデータの配布はIPマルチキャストを使用したプッシュ型配信も有効である。このようなプッシュ型配信では、受信ホストの使用条件や用途により送信データの種類毎に受信ホストをグループにする必要がある。本稿では、IPマルチキャストを使用したデータ配信におけるグルーピング制御を行なう簡単なプロトコルについて述べる。

IP multicast grouping protocol

Akira Shinoda

poripori@slab.ntt.co.jp

NTT Software Laboratories

Abstract

We can use IP multicast systems to communicate with several hosts simultaneously. A group is defined on IP multicast communication systems. While Audio/Video broadcasting is popular on the MBone, the use of groups can be very effective for push data for software delivery and data delivery. In the case of push delivery, It is necessary to group receivers of pushed data according to their environment or needs. In this paper, we describe simple grouping control protocol for data delivery using IP multicast.

1. はじめに

同時に複数対向と通信する方法として同報通信がある。Internet Protocol(IP)では、RFC1054[1]で IP の拡張仕様として IPマルチキャストが定義されている。このIPマルチキャストを用いることにより ネットワーク層では複数対向への通信が可能となる。

IPマルチキャストでは複数対向への通信の範囲としてグループを形成することができる。これはユニキャストの送信時に宛先アドレスとして相手となるホストのA/B/CのいずれかのクラスのIP アドレスを指定する代わりに、マルチキャストではD クラス IP アドレスを指定し、受信時には、その D クラス IP アドレスを検出することでグループを形成する。これにより、グループに属するホストのみ データを受信することが可能となる。

現在、IP マルチキャストの世界規模のネットワークであるMBoneでは、セッションを作成して通信を行っており、そのセッション内のメディア単位ではグループを形成している。データはオーディオ/ビデオが中心であり、ラジオ/テレビの周波数/チャンネルのように IPマルチキャストのグループが使用されている。

オーディオ/ビデオのように放送のような利用形態であれば、受信側で、そのグループの D クラス IP アドレスに合わせれば受信が可能となる。一方、プログラムの配布、データファイルの配布などをプッシュ配信する場合は、送り側主導となる。つまり、放送型の通信形態であれば、受信者の意志によってグループを取捨選択し希望のグループを受信するのに対し、一方向のプッシュ型通信形態では、あらかじめ決められた受信先をグループ化して通信する必要が出てくる。例えば、数百台のマシンのアプリケーションを一手にバージョン更改する場合、使用する用途などによりアプリケーションが違うため、アプリケーション種別毎にグループの作成/削除、グループへの参加/離脱の制御や管理が必要となり、これらの作業は送信元で行なうこととなる。

本稿では、人手による操作がない受信ホストに対してIPマルチキャストを使用したデータ配信におけるグルーピング制御を行なう簡単なプロトコルについて述べていく。

2. グループを単位とした通信

通信形態の定義として、不特定多数への通信をブロードキャスト、1対1通信をユニキャストと呼んでいる。

TV放送はブロードキャストであり、電波としてすべての受信機で受信される。希望の周波数をユーザーが選択するが、配信形態としてはプル型であり、すべての人が受信していることになる。

インターネットで、ユーザーが Web サーバーにアクセスしてブラウザで画面に表示させる場合、HTTPを使用するが、これは1対1の通信でユニキャストである。

IPマルチキャストを使用した場合、送信側の D クラス IP アドレスによりグループが形成される。受信側では、送信側の D クラス IP アドレスに参加(Join)することにより受信することができる。このような、特定グループを形成して通信することをグループキャストと呼ぶこととする。

3. MBoneでのセッション指向な通信

MBone におけるグループへの参加手順について述べる。

現状の MBone では一般的には以下の手順によりグループを形成しグループによる送受信を行なっている。

- (1) 送信側でセッションを作成する。
- (2) 送信側からセッションディレクトリ[2][3]によりアナウンスする。
- (3) 受信側はセッションディレクトリを参照する。
- (4) 受信側はセッションディレクトリより希望のセッションを選択する。

セッションはプログラム(番組)の単位であり、オーディオ/ビデオ/共有ウィンドウなどの複数のメディアを含んでおり、複数のグループを所有している。また、セッションディレクトリのアナウンス

メントとメディア（実際の通信データ）は分離されており、アナウンスはアナウンスのみとなっている。上記の手順のように、受信側の意志によりマルチキャストグループへ参加することを本稿ではアクティブ参加と定義する。

4. グルーピング制御

3章で述べたアクティブ型参加に対し、受信側での人手による操作がないプッシュ型通信の形態でのマルチキャストグループへの参加をパッシブ参加と定義する。パッシブ参加では、送信側で作成したグループに対し受信側にそのグループへの参加をする／しないを認識させる方法が必要となる。つまり、グループの形成／解散を制御するグルーピング制御を送信側から行なうことになる。

3章におけるセッションディレクトリによるアナウンスを使用した方法では受信側ではセッションリストを参照することによりセッション（グループ）への参加を行なっている。これを1つのパラダイムとすると、同様にパッシブ参加でもアナウンスにより受信側に参加するグループ名のリストを持たせて参照させる方法が考えられる。この場合、アナウンスとしての機能は、単純なアナウンスではなく、受信側でのグループ名リストを操作する機能を持たせることになる。つまり、リストに入っている／いないによってグループへ参加する／しないとなるため、これらの操作を制御する。

データ配送、ファイル配送はオーディオ／ビデオの放送型の場合と違い、常にデータが送信されていることはなく、特定の時間に送信される。そのため、常に受信しているという状態は不要となる。そこで、アナウンスは、送信開始をアナウンスし受信側で予めグループに参加し受信動作準備をさせる機能をもつことになる。

5. グルーピング制御プロトコル(Grouping Control Protocol:GCP)

4章よりグルーピング制御に必要な機能は、送信側が受信側にグループへ参加する／しないを認識させるためのリストの操作機能とグループへ参加をさせる機能であることがわかった。

リストの操作は、グループ名の追加／削除を行なう。つまり、リストでのグループ名のある／なしによってグループに属する／属さないという意味である。

送信側からのあるグループに属する／属さないの制御はあるグループ名に対して参加ホストのIDリストを送ることとした。これは個別に通信するのではなく、アナウンスで行なうためである。受信ホストでは、そのリストに自ホストのIDが存在するかどうかをチェックすることになる。識別子は受信ホストがユニークに識別できるものであることが必要である。そのため、IPアドレスとFQDNが考えられる。今回は、4バイト固定長で処理が簡単になれるためIPアドレスを使用した。

グループへ参加させる機能は、送信側がこれからデータ送信するというグループに受信側を参加させる操作である。そのため、まずグループ名をアナウンスし受信側で自ホストが該当するグループに属しているかをチェックさせる必要がある。次に受信側では該当するグループであればグループに参加する。その際に必要な情報は、グループのIPアドレスとポート番号である。これは受信側ではグループに対するそれらの情報を持たず、識別子としてのグループ名だけを持つようにし、それらの情報も送信側から通知するようにする。グループのIPアドレスはWellKnownのものを除いて不定であり、あるIPアドレスを使用しようとした時に既に使用されている可能性がある。通信上重複した状態を避けるなら、特定の送信ホストのみを識別する方法やポート番号を変更する方法もあるが、IPマルチキャストグループという単位を考えた場合、IPアドレスを他のグループと重複させない方が望ましい。そのため、送信側でのIPアドレス使用状況のチェックや選択が可能なように、送信側からグループのIPアドレスとポート番号も通知する。

以上のことより送信側から送るメッセージの機能は以下の3つとなる。

- (1)リストへのグループ名の追加
- (2)リストからのグループ名の削除
- (3)グループへの参加

これらの機能のメッセージフォーマットを図1に、メッセージで使用するオペレーションコードを

表 1 に示す。上記(1)(2)(3)の機能がオペレーションコード 8、9、7 に対応する。

基本は、オペレーションコード、ポート番号、IPアドレス、データの並びであり、ポート番号は、RDY の場合はポート番号、アドレスリストの場合は、アドレスの数を示すようにした。また、グループ名は可変長データであるため、固定長データの後ろに並べヌル終端にした。受信ホストでの動作完了をチェックするためAcknowledgmentも用意し、これのみとした。通信シーケンスを図 2 に示す。

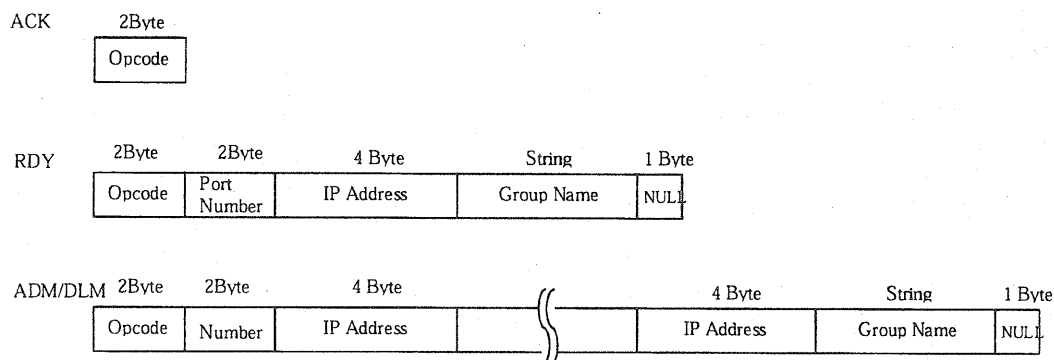
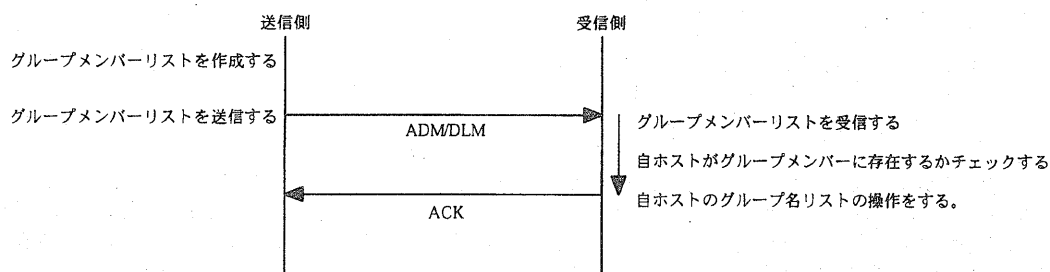


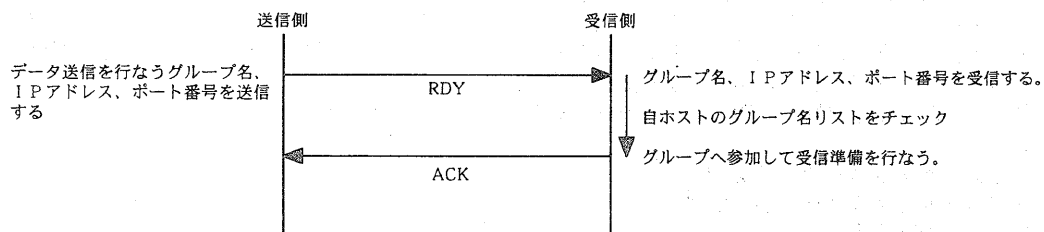
図 1 メッセージフォーマット

表 1 オペレーションコード

opcode	operation
4	Acknowledgment (ACK)
7	ready to transfer data(RDY)
8	add members to a group(ADM)
9	delete members from a group(DLM)



(1)(2) リストに対するグループ名の追加／削除



(3) グループへの参加

-図 2 通信シーケンス

メッセージに対する応答をACKのみにしたのは、前述の処理が1メッセージに対する処理であり、メッセージに対して正常処理が行なわれたことのみを検出できればよいからである。処理が未遂行のためにNACKを返した場合とメッセージ不達もしくは受信側ビジーによりNACKを返さない場合は処理が未遂行であることは同様であるため、受信側の処理完了の判断はACKもしくはそれ以外ということで判断できる。

6. 実装

今回の実装では、UNIX系のOSであるFreeBSD、NetBSD、Solaris、SunOS上で行ない、送信側をコマンドとして、受信側をデーモンとして作成した。また、本プロトコルのみであるとグルーピングの制御のみになるので、TFTPを使用してデータ転送機能も実装した。ただし、本プロトコルおよびTFTPにおいてのACKメッセージは、ACK受信側でのオーバーフローについては未検討であるため実装を行なわなかった。そのため、TFTP部分ではフロー制御が不可能なので、受信バッファが満杯にならない大きさのファイル転送しかできないという条件になった。実装したツールで使用するグループは、グルーピングの制御を行うためのアナウンス用のグループとデータ転送用のグループの2種類とし、データ転送用のグループはグループごとにIPアドレスをもつことができるようにした。グルーピング制御のみの動作フローを図3に示す。

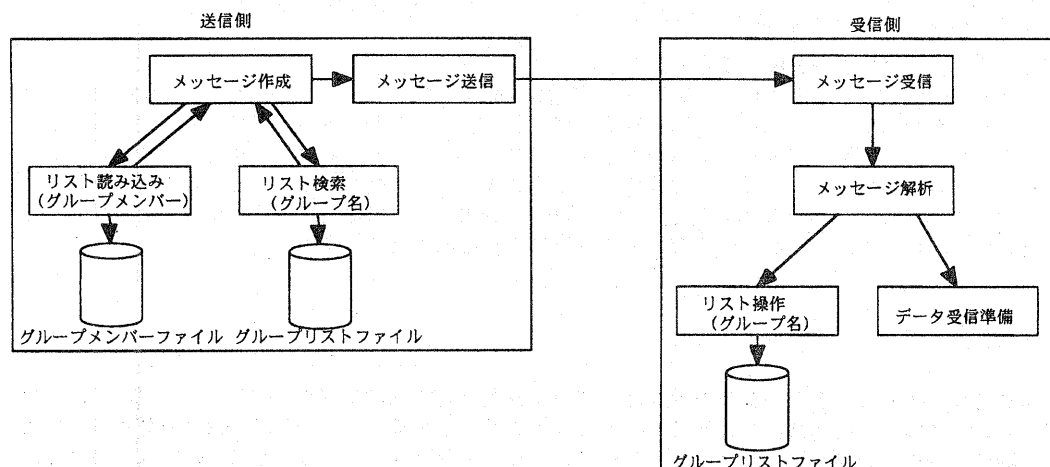


図3 グルーピング制御の動作フロー

送信側では、グループメンバーファイルとグループリストファイルを持たせた。グループメンバーファイルにはグループに属する受信ホストのIPアドレスとホスト名を/etc/hosts形式で記述する。グループリストファイルでは、グループ名、IPアドレス、ポート番号、TTLを記述する。受信側では、グループリストファイルを持たせた。このファイルに属しているグループ名のみを記述していく。それぞれのファイルの中身をリスト1～3に示す。受信側での識別はグループ名のみであり、データ配送前にメッセージとしてIPアドレスを送信するので、リスト2のように固定的に決定しなくても、動的にIPアドレスを割り当てることも可能である。

また、データ受信はRDYメッセージを受信した後でデータ転送用のグループに参加(Join)し、受信動作に移るようにした。受信後、後処理としてグループからの離脱(Leave)を行なっている。

7. 考察

受信ホストの識別子として今回はIPアドレスを使用した。DHCPでアドレスが動的に割り振られていくことが多くなった最近では、ユニークな識別子としての意味がなくなってくる。そのため、

データとしては可変長になるが、FQDNを使用した方が識別子として有効であると思われる。

本方式では受信ホストがグループに所属する／しないをメッセージとして予め通知し、受信ホスト内のファイルとしてグループ名を保存し、グループへ参加する場合もメッセージとしてグループ名を送り、そのグループ名をチェックしてIPマルチキャストグループへの参加を行なう。もう1つ考えられる方法として、データ送信前の受信ホストのチェックの機能も含め受信ホストのリストを送り全受信ホストでチェックさせ、自ホストが該当すればグループに参加するという方法である。後者の場合、データ送信直前に受信ホストリストのチェックを行うため、受信ホスト数が増加した場合、送信メッセージが大きくなることと、受信直前にリスト検索を行うため、受信処理に移るまでに時間を要する。本方式では、所属グループの認識と実際のグループへの参加処理を分離している。そのためオペレーションとしてグループの

管理作業とデータ配送を分離することができる。

プロトコルではACKも定義した。送受信1対複数の通信の場合、ACK受信時のバッファ溢れを引き起こす可能性がある。このバッファ溢れは、ソケットバッファで起こることが確認されている[4]。GCPでのACKは2バイトであり、UDPヘッダーを含めて10バイトとなる。通常、UDPバッファは4kバイト程度のことが多く、この場合でも400メッセージを受け取ることができる。また、ACKに対する処理が逐次行なわれること、ルーター、レピーターではパケットがシリアルライズされることを考えれば、1,000台分程度のACKは処理できるものと予測する。

8. おわりに

本稿では、IPマルチキャストを使用したプッシュ配信時におけるグルーピングを制御する簡単なプロトコルについて述べた。SDR[5]でのセッション情報とは違い、サービスに関する情報を持たせず単純にグルーピングを行なうためだけのものとなった。今後は、ACK応答については未検討であるため、検討を進めると共にリライアビリティを持ったマルチキャスト転送プロトコルを使用し、1つのマルチキャストファイル転送ツールとして制作していく予定である。

謝辞

本研究の機会を与えていただいた、NTTソフトウェア研究所情報流通基盤プロジェクトチーム川野邊正リーダー、基礎技術研究グループ中村雄三リーダー、武藤信夫主任研究員に記して感謝する。

【参考文献】

- [1] S. Deering, "Host Extensions for IP Multicasting", RFC1054, Stanford Univ., May 1988
- [2] M. Handley and V. Jacobson, "SDP: Session Description Protocol", INTERNET-DRAFT, 22nd Jan 1997
- [3] M. Handley, "SAP - Session Announcement Protocol", INTERNET-DRAFT, November 25th 1996
- [4] 城下他, "高信頼マルチキャスト通信プロトコル(RMTP)の各種ネットワークへの適用性", 信学会 IN 研究会 信額技報IN95, Mar 1996.
- [5] <http://north.east.isi.edu/sdr>

172.22.16.196 red
172.22.19.183 wonder
172.22.19.184 chip
172.22.20.38 pecker
172.22.20.183 yellow
172.22.20.218 quad

リスト1 送信側グループメンバーファイル

all	239.2.2.2/23567/16
tokyo	239.2.2.4/23567/16
chiba	239.2.2.6/23567/16

リスト2 送信側グループリストファイル

all
tokyo

リスト3 受信側グループリストファイル