カメラ PC と IP マルチキャスト

岡村耕二　柴田賢介

1　はじめに

本稿では、水泳大会における IP マルチキャスト (以下、マルチキャストと表記する。) ネットワークと、各会場に数箇所設置したカメラ PC からの動画像マルチキャストについて報告する。また、Mhealth を用いて、マルチキャストネットワークルータや送信ホスト、受信ホストおよびマルチキャスト配送のためのツリー情報を自動獲得した結果についても報告する。

2　マルチキャストネットワークの構成

マルチキャストネットワークを設計する場合、まず、マルチキャスト経路制御プロトコルを決定する必要がある。今回は、各会場のコアルータに FreeBSD が載った PC ルータが用いられていた点と、今回使用するカメラ PC からの動画像をマルチキャストで受信するホストが各会場で非常に高い確率で存在するであろうことを予測して DVMRP を用いた。

QGPOP の九州大学、財団法人 九州システム情報技術研究所 (以下、ISIT と表記する)、天神の各 NOC のコアルータは FreeBSD が載った PC ルータおよび GR2000 が設置されている。各会場のコアルータと GR2000 とはバックボーン用に PVC が直接張られているし、また、ルータとしてはパケットの処理能力も高いため、各会場からのマルチキャストの接続には GR2000 を使用する計画であった。しかし、GR2000 の DVMRP の実装には mtrace の機能がないことが途中で判明したため、GR2000 は今回のマルチキャストの実験では利用しないことにした。GR2000 の代わりには各 NOC にある FreeBSD の載った PC ルータを用いた。しかし GR2000 から PC ルータに変更した時、各会場の PC ルータと NOC の PC ルータ間の PVC は用意されていなかったため、まず、トンネリングでマルチキャストネットワークを構成し、様子を見て性能が悪いようであれば、直接 PVC を張ることにした。結果、カメラ PC からの 2Mbps 程度のマルチキャストトラフィックのみであればトンネルでも性能的な問題はないようであったので、PC ルータ同士をトンネリングで接続させてマルチキャストネットワークを構築することはかなり古い方法であることは認識した上でそのままトンネルで運用した。トンネリングは最下層の ATM メガリンクのトポロジーと対応させて、マリンメッセと天神 NOC、博多の森および県営プールと九州大学、西市民プールと ISIT とそれぞれトンネル接続させた。

3　カメラ PC からの動画通信

カメラ PC はいわゆるコンパクト PC タイプの筐体の PC にカメラを接続し、各会場の無線機器などのそばに設置しそこで入力した動画をネットワークに通信させるものである。OS として Linux を搭載させ、カメラデバイスとして Creative 社製の WebCamIII (図 1)) を利用した。WebCamIII は USB インタフェースで接続するタイプで、Linux からは Video for Linux API で利用することができる。

図 1: WebcamIII

動画の通信には筆者が独自に開発した JPEG 形式にエンコードした画像を RTP で通信するツール pachapo-cam を用いた。JPEG over RTP は、RFC2035 互換の形式なので受信者は vic で受信することができる。図 2 に受信者側で vic を起動し、各会場から pachapo-cam で送信された動画を受信しているところ示す。pachapo-cam はコマンドラインで実行でき、また、フレームレート、JPEG の Q Factor などコマンドのオプションで指定することができるため、遠隔にあるカメラ PC からの動画通信を容易に制御することができた。pachapo-cam

を利用せずに例えば、vic をカメラ PC で起動させ大学などの手元にあるホストの X Window System に張り付ける方法も考えられたが、途中のネットワーク障害による再起動が困難であり、また、途中のネットワークの輻輳状態に依存して Window のボタンやスクロール操作がやり難くなる問題があった。

また、pachapo-cam はスクラッチから筆者がコーディングしたものであったため、機能拡張が容易で、例えばカメラから入力した画像に文字のコメントを張り付けたり (図 3)、他の方法で生成させた JPEG ファイルをそのまま RTP で送信する (図 4) ことなどが自由にできた。

図 3: 動画にコメントを張り付ける

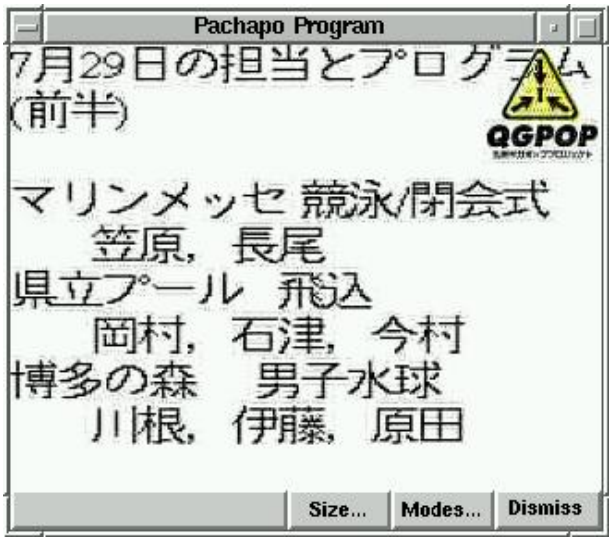

図 4: 文字情報を静止画として送る

QGPOP は、現在、IMnet および九州大学を経由して Mbone と接続しているが、今回の実験のトラフィックは QGPOP 内に閉じるよう pachapo-cam からの初期 TTL 値を 31 に設定した。しかし、初期 TTL 値が 31 であると QGPOP から九州大学にもマルチキャストパケットは配送されない。今回九州大学内にも受信を希望するホストが多数存在していたため、実験期間中のみ、九州大学と QGPOP の間の Threshold 値を 16 に変更して運用した。1 台のカメラ PC から送出されるトラフィックは 100〜300Kbps で、マルチキャストの総トラフィックは約 2M bps 程度であった。図 5にマリンメッセに設置していたカメラ PC の通信ログを示す。塗りつぶされているのが受信パケットで、実線が送信パケットである。なお、図 5はイーサインタフェースの送受信パケットを SNMP で記録したものであるため他のトラフィックログも混ざっているが、ほとんどは動画のマルチキャストによるトラフィックである。pachapo-cam は指定されたマルチキャストアドレスに join するためプログラム中ではパケットを受信しないけれどもインタフェースには受信ログが残る。

オープンウォータースイミング競技の開催されたももち浜のプレス室は、無線で接続されており回線速度は潤沢でなかったため、ももち浜はマルチキャストで接続しなかった。ももち浜に設置したカメラ PC からは pachapo-cam で帯域を 100Kbps 以下になるよう JPEG の Q Factor を調整し、ユニキャストで一旦九大まで送信し、九大でそのユニキャスト RTP パケットをマルチキャスト RTP に変換してマルチキャストネットワークに流し込んだ。また、福岡タワーに設置していたコンピュータ制御可能なカメラは筆者の開発した vvideo (Virtual Video Capture) デバイス、および、冷水氏の開発している Union Camera を利用して九大に設置した Union Camera Client で画像をキャプチャしマルチキャストネットワークに流し込んだ (図 6)。カメラはパンを左右に自動的に動かしたり、ズームを変化させていたがこれらの制御も九大に設置していた Union Camera Client で行なっていた。

4 MHealth を利用したマルチキャストトポロジ獲得実験

今回、我々は福岡で開催された世界水泳選手権の各会場に、機器監視用のカメラを設置し、このカメラから得られる画像をマルチキャストで配信するという実験を行なった。このマルチキャストでの画像配信を利用し、MHealth を用いたマルチキャストトポロジの獲得実験を同時に行なった。本章では、この実験の概要と結果について述べる。

図 2: vic を使って受信している様子

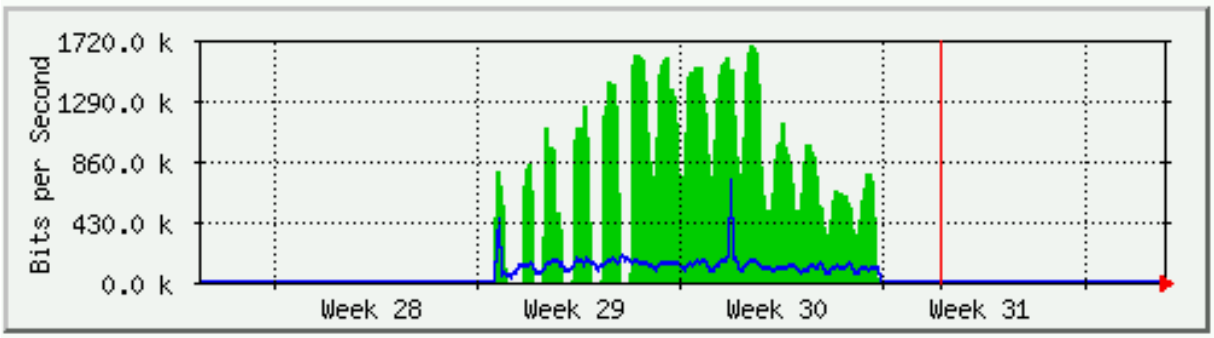

図 5: マリンメッセのカメラ PC の通信ログ

図 6: 福岡タワーからの映像

4.1 実験の概要

今回の実験では、各会場 (5 カ所) にそれぞれ 1〜2 台のカメラを設置し、カメラから得られる画像をマルチキャストで配信するプログラムによって送信した。これに対し、受信者は各会場、そして九州大学内で vic という動画像を送受信するためのツールを利用して受信した。この配信プログラムと vic は共に RTCP に対応しており、送受信者間でレポートの交換が行なわれる。受信者は各会場 (QGPOP セグメント内に位置) と九州大学に拡がっており、送受信者ともにデータや制御パケットを送信する際の TTL 値は 31 とした。

4.2 実験結果

前節で述べたような実験環境において、マルチキャスト通信を行なっている時に、MHealth を実行した結果を図 7として示す。今回 MHealth は QGPOP セグメントから実行している。この図において、上部に並んでいるのが送信者であり、下部が受信者である。図中左端の送信者 (133.69.169.7) から全ての受信者への配送木が描かれている。また、図 8は受信者の中の 1 人について、パケットロス率や mtrace の状況などを表示した画面の一部である。図 8では上部に RTCP から得られた受信者情報と、mtrace の進行状況などが書かれており、下部には RTCP から得られたパケット損失率の情報が書かれている。

今回の実験ではテキスト形式のログの出力も行なった。このログの一部を付録 1、2 として示している。付録 1 が RTCP から得られたデータのログであり、付録 2 は mtrace から得られたものである。RTCP のログからは、データを取り始めた時間を基準としたタイムスタンプや、RTCP のレポートの送信元、パケットロスの数やジッタなどの情報を得ることができる。mtrace のログを見ると、MHealth はまず送信者から受信者への mtrace を行ない、失敗した場合には MHealth を実行しているホストから送信者、受信者への mtrace を行なってトポロジ情報を獲得していることが分かる。

今回、我々は世界水泳選手権の会場に設置したカメラを利用して、マルチキャストのデータ配信と、データの配信の際のマルチキャストの配送木の情報を MHealth というツールを用いて獲得するという実験を行なった。結果は、図 7に示した通り、送信者から受信者への配送木を描くことができた。しかし、MHealth には以下のような問題点があることも実験を通して分かった。

1. mtrace の実行が終了するまでに多くの時間を要する。
2. 各受信者に対して mtrace をそれぞれ実行するため、規模適応性に欠ける。
3. 複数の送信者が存在する場合の対応がまだできていない。
4. ルータになっているホストが受信者にもなる場合、配送木が正しく描けない。

(1)、(2) は非常に大きな問題点であり、より大きなグループに対してトポロジ情報の獲得を試みる際には別の手段を考える必要がある。以上のような問題点はあるが、MHealth は結果がグラフィカルに表示されるため、分かりやすく操作が簡単であるという利点もある。今後はネットワーク管理以外にも、研究などの分野において利用される場面も出てくるであろう。

5 おわりに

今回のカメラ PC を用いたマルチキャスト通信実験は、比較的クラシックな技術を使ったため概ねうまくできた。GR2000 の DVMRP が mtrace を使えないのは以外だった。今後サポートの予定もないらしいため QGPOP のマルチキャストで GR2000 が使われることはないだろう。多数のホスト、ルータがいる状態で Mhealth を実行させたのは今回が初めてであったが、本文であげているように Mhealth の問題を確認することができた。

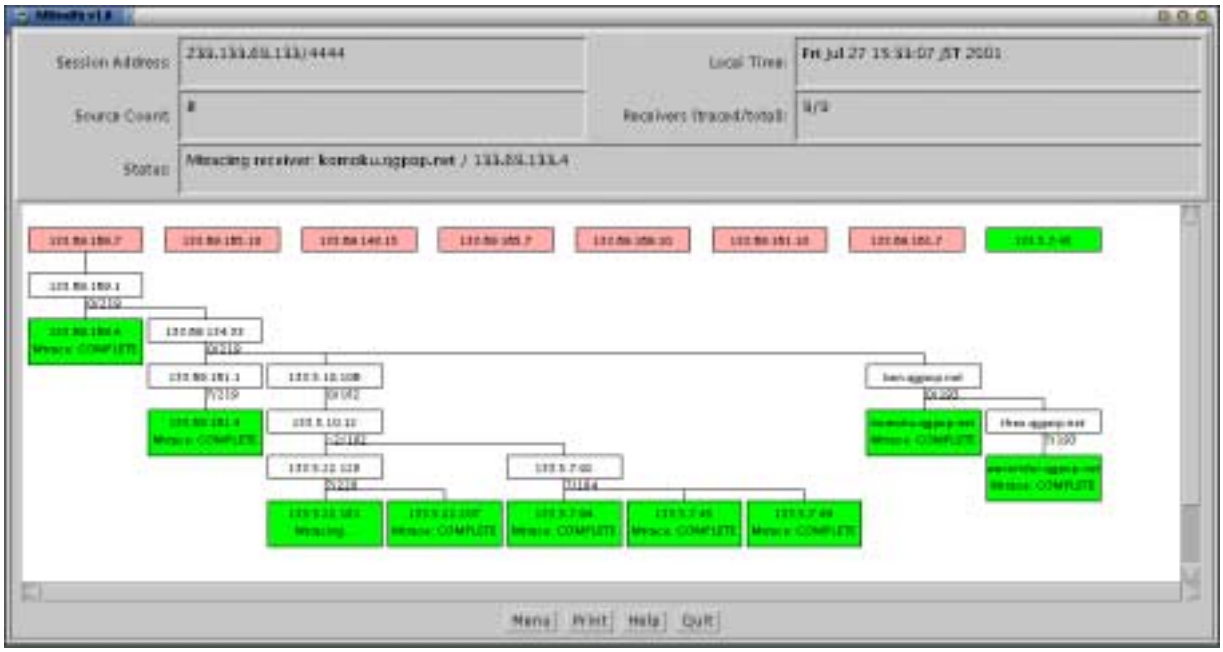

図 7: MHealth の実行結果

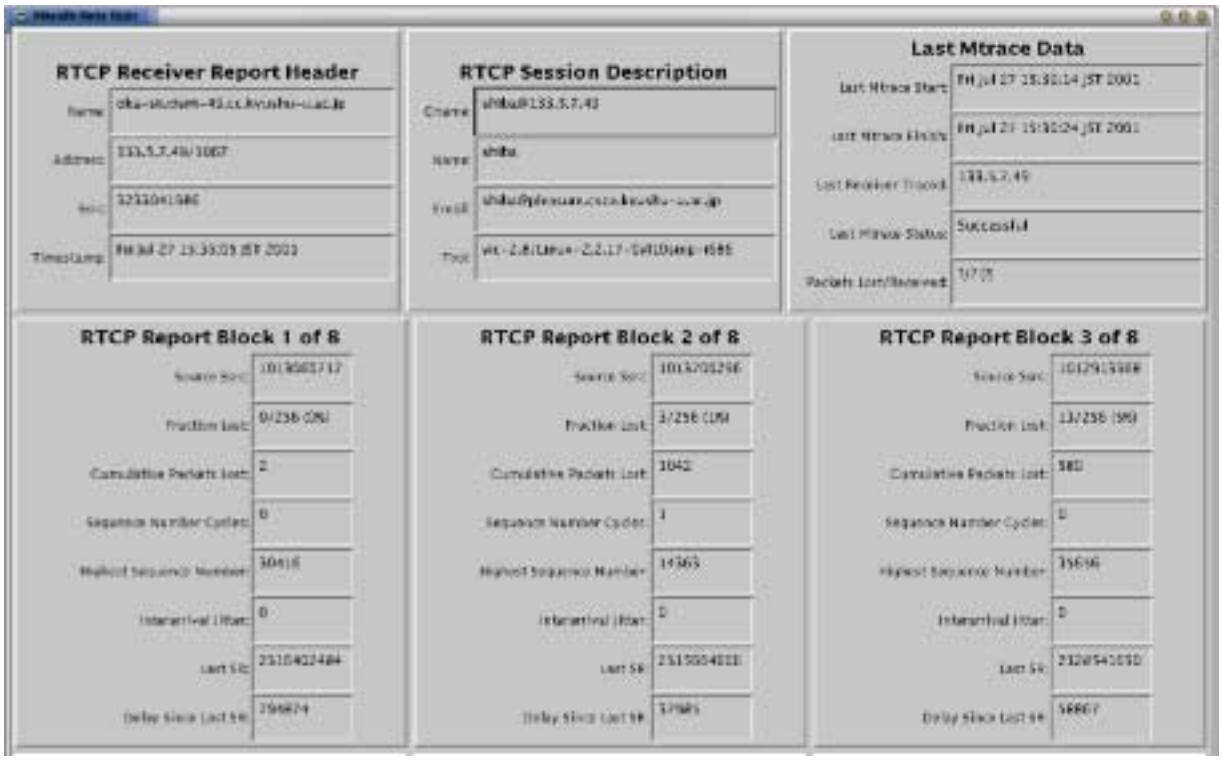

図 8: 受信者の状況を示した画面

付録 1：MHealth のログ (RTCP)

```
# RTCP PACKET LOG
# LOG SESSION: 239.133.69.133/4444
# LOG START TIME: Fri Jul 27 15:23:57 JST 2001 = 996215037097
# LOG FORMAT: 1-timestamp 2-ip 3-ssrc 4-reportcount n-fractionlost#n n-jitter#n
996215037097 133.69.169.7 1013180266 0
996215037487 133.69.165.10 1013000246 0
996215040136 133.69.148.15 1013378005 0
996215041748 133.69.169.44 3232368941 8 85 0 64 0 72 0 54 0 70 0 71 0 70 0 67 0
996215042109 133.5.7.49 3233041986 8 0 0 0 0 0 0 5 0 5 0 9 0 0 0 0 0
996215042946 133.69.169.4 992755505 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215043051 133.69.165.7 1012913968 0
996215043347 133.69.169.10 1013700296 0
996215043733 133.69.161.10 1013312905 0
996215043794 133.69.161.7 1013276427 0
996215043827 133.69.169.7 1013180266 0
996215043859 133.69.165.10 1013000246 0
996215043903 133.69.148.15 1013378005 0
996215044153 133.69.161.4 992358201 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215044739 133.5.22.181 3232201922 7 8 0 10 0 0 0 12 0 56 0 9 0 18 0
996215044766 133.69.165.7 1012913968 0
996215044809 133.69.169.10 1013700296 0
996215044843 133.69.161.10 1013312905 0
996215044887 133.69.161.7 1013276427 0
996215044935 133.69.169.7 1013180266 0
996215045081 133.69.133.4 3232938360 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215045101 133.69.165.10 1013000246 0
996215045748 133.5.7.48 1013060717 0
996215045768 133.69.148.15 1013378005 0
996215045797 133.69.165.7 1012913968 0
996215045843 133.5.7.48 3231484045 8 13 0 3 0 14 0 10 0 16 0 14 0 44 0 0 0
996215045853 133.69.169.10 1013700296 0
996215045908 133.69.161.4 992358201 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215045952 133.69.169.10 1013700296 0
996215046084 133.69.133.19 3232738707 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215046105 133.69.169.7 1013180266 0
996215046120 133.69.161.7 1013276427 0
996215046135 133.69.169.44 3232368941 8 46 0 60 0 79 0 59 0 71 0 68 0 68 0 64 0
996215046151 133.69.169.7 1013180266 0
996215046186 133.69.165.10 1013000246 0
996215046204 133.69.165.7 1012913968 0
996215046308 133.69.133.4 3232938360 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215046327 133.69.161.4 992358201 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215046337 133.69.169.10 1013700296 0
996215046370 133.69.161.10 1013312905 0
996215046402 133.69.161.7 1013276427 0
996215046433 133.69.169.7 1013180266 0
996215046459 133.69.165.7 1012913968 0
996215046500 133.69.169.10 1013700296 0
996215046527 133.5.7.48 3231484045 8 7 0 0 0 2 0 9 0 21 0 9 0 6 0 0 0
996215046545 133.69.161.10 1013312905 0
996215046581 133.69.161.7 1013276427 0
996215046618 133.69.169.7 1013180266 0
996215046652 133.69.169.4 992755505 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215046672 133.69.165.10 1013000246 0
996215046717 133.69.148.15 1013378005 0
996215046743 133.69.165.7 1012913968 0
996215046880 133.5.22.207 996447471 8 0 0 3 0 0 0 12 0 0 0 10 0 2 0 21 0
996215046902 133.69.169.10 1013700296 0
996215046924 133.69.161.10 1013312905 0
996215046961 133.69.133.19 3232738707 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
996215046988 133.5.7.49 3233041986 8 0 0 18 0 4 0 16 0 11 0 5 0 8 0 2 0
996215047003 133.69.161.7 1013276427 0
996215047063 133.69.169.44 3232368941 8 69 0 68 0 70 0 64 0 80 0 71 0 74 0 75 0
```

付録 2：MHealth のログ (mtrace)

```
# ROUTE LOG
# LOG SESSION: 239.133.69.133/4444
# LOG START TIME: Fri Jul 27 15:24:01 JST 2001 = 996215041725
#####################################################################
# TIMESTAMP START = 996215041726

mtrace 133.69.169.7 133.69.169.44 239.133.69.133
Mtrace from 133.69.169.7 to 133.69.169.44 via group 239.133.69.133
Querying full reverse path... * switching to hop-by-hop:
  0  pachapo-dhcp-avispa-169-44.qgpop.net (133.69.169.44)
 -1  * * *
 -2  * * *
 -3  * * *
 -4  * * * ...giving up
Timed out receiving responses
Perhaps receiver 133.69.169.44 is not a member of group 239.133.69.133,
or no router local to it has a route for source 133.69.169.7,
or multicast at ttl 127 doesn't reach its last-hop router for that source
# TIMESTAMP FINISH = 996215080796

# TIMESTAMP START = 996215080798

mtrace 133.69.169.44
Mtrace from 133.69.169.44 to 133.69.134.34 via group 0.0.0.0
Querying full reverse path...
  0  toy.qgpop.net (133.69.134.34)
 -1  okalab-students.qgpop.net (133.69.134.33)  DVMRP  thresh^ 1
 -2  pachapo-avispa-router.qgpop.net (133.69.169.1)  DVMRP  thresh^ 1
 -3  pachapo-dhcp-avispa-169-44.qgpop.net (133.69.169.44)
# TIMESTAMP FINISH = 996215080881

#####################################################################
# TIMESTAMP START = 996215080881

mtrace -g 133.69.169.1 133.69.169.7 239.133.69.133 133.69.169.44
Mtrace from 133.69.169.7 to 133.69.169.44 via group 239.133.69.133
Querying full reverse path...
  0  pachapo-dhcp-avispa-169-44.qgpop.net (133.69.169.44)
 -1  pachapo-avispa-router.qgpop.net (133.69.169.1)  DVMRP  thresh^ 1  Wrong interface
 -2  pachapo-cam-11.qgpop.net (133.69.169.7)
Round trip time 1 ms; total ttl of 2 required.

Waiting to accumulate statistics...
# TIMESTAMP FINISH = 996215082554

#####################################################################
# TIMESTAMP START = 996215122082

mtrace -g 133.69.169.1 133.69.169.7 239.133.69.133 133.69.169.4
Mtrace from 133.69.169.7 to 133.69.169.4 via group 239.133.69.133
Querying full reverse path...
  0  pachapo-avispa-demopc.qgpop.net (133.69.169.4)
 -1  pachapo-avispa-router.qgpop.net (133.69.169.1)  DVMRP  thresh^ 1  Wrong interface
 -2  pachapo-cam-11.qgpop.net (133.69.169.7)
Round trip time 1 ms; total ttl of 2 required.

Waiting to accumulate statistics...Results after 10 seconds:

  Source         Response Dest    Overall     Packet Statistics For Traffic From
133.69.169.7     224.0.1.32       Packet      133.69.169.7 To 239.133.69.133
     v       __/  rtt    1 ms     Rate        Lost/Sent = Pct  Rate
133.69.169.1     pachapo-avispa-router.qgpop.net Wrong interface
     v       \__  ttl    2         93 pps        ?/218          21 pps
133.69.169.4     133.69.134.34
  Receiver       Query Source

# TIMESTAMP FINISH = 996215132175
```