

1. はじめに

多様なトラフィックが IP ネットワーク環境に統合されつつある。このような状況においてアプリケーション利用環境の向上を図るには単にネットワークの高速化だけでなく、トラフィックフローあるいはフローの集合に対してサービス品質 (QoS) を制御する機構が望まれる。これに関して RSVP (Resource Reservation Protocol), Diffserv (Differentiated services) など網資源予約, 優先制御に関わる諸技術が IETF (Internet Engineering Task Force) にて検討されている。これらの技術をネットワークに導入する場合, 網資源確保の乱用を防ぐために優先呼受付基準や網資源割当て量の制約など管理者の QoS ポリシを対象ネットワークの各ノードに反映させる必要がある。このような課題の対処法の 1 つとして, ポリシに基づくネットワーク機器/網資源の集中管理技術が検討されている。筆者等は RSVP 網における網資源をポリシーベースで集中管理する簡易システムを開発した。本稿ではその実装と性能評価について報告する。

2. システム構成

開発システムの構成を図 1 に示す。本システムは RSVP 網に対して適用され, 同プロトコルによって転送される各種 RSVP オブジェクトの情報をシステムの管理, 制御の対象としている。

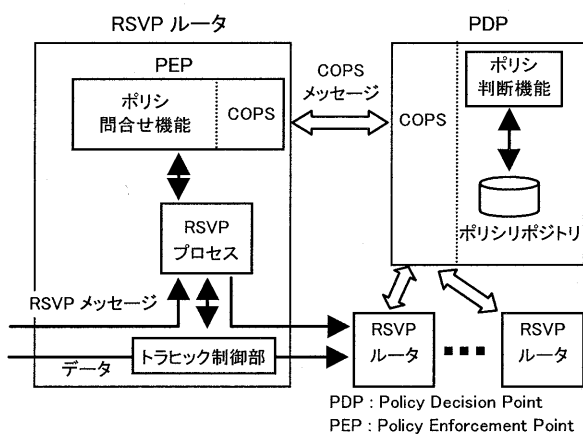


図 1. システム構成

“Policy Based Simple QoS Management System” by Osamu MAESHIMA, Masami ISHIKURA, Toshihiko KATO and Tohru ASAMI, KDD R&D Laboratories Inc.

システムは (1) 筆者らが開発した RSVP ルータ[1] 上に実装し RSVP メッセージ処理に関してポリシー判断要求を行うポリシー問合せ機能, (2) RSVP ルータあるいは外部ノード上に実装しポリシー判断を実行するポリシー判断機能, (3) ポリシメッセージ転送用プロトコル (COPS), (4) ポリシリポジトリより構成される。

2.1. ポリシ問い合わせ機能

RSVP プロセスが保持していない新規の RSVP セッション用メッセージあるいは RSVP オブジェクトの内容が変更された既存 RSVP セッションのメッセージは RSVP プロセスより PEP (Policy Enforcement Point) へ渡される。PEP は RSVP プロセスの処理内容 (メッセージ受信, トラフィック制御設定, メッセージ転送), RSVP オブジェクトの情報等を PDP (Policy Decision Point) へ転送し, メッセージ処理判断を依頼する。また, PDP からの応答内容に基づき RSVP プロセスに対して当該 RSVP メッセージの処理について指示 (許可/棄却) を与える。

PEP と PDP 間のメッセージ転送には COPS (Common Open Policy Service) プロトコル[2] を採用し, 筆者らが開発した RSVP ルータおよび PDP となるノード (OS: FreeBSD release 2.2.5 以上) に実装した。

2.2. ポリシ判断機能

COPS セッション経由で PEP から受信するポリシー判断要求に対して応答する。PDP は起動時にポリシリポジトリ内のポリシルール定義ファイルを読み込み, PEP からの問い合わせ内容をこれと照合して判断を下す。

2.3. ポリシリポジトリ

管理ネットワーク全体にわたる統一されたポリシルールの集合を保存するためのデータベースである。IETF POLICY-WG ではポリシー情報を表現するため DMTF (Distributed Management Task Force) で定義する共通情報モデル (CIM: Common Information Model) の一部としてポリシオブジェクトクラス構造を構成し, さらに LDAPv3 などのディレクトリアクセスプロトコル用オブジェクトへのマッピングを検討している。

筆者等はオブジェクトクラスの構造と UNIX ファイルシステムのディレクトリ構造が良い対応関係にあることに着目し, ここではポリシオブジェクトクラス定義は踏襲しつつ, ディレクトリアクセス部分を UNIX ファイルシステムへのアクセスに置き換えることで実装の簡易化を図った。すなわち, ポリシリポジトリ部分を ASCII ファイルの集合で表現し, ポリシクラス階層を UNIX ファイルシステムに展開

する方式を採用した。

3. 評価実験

RSVP メッセージの転送時間に着目して開発システムの性能評価を行った。

3.1. 実験構成

図 2 に実験構成を示す。PDP と PEP を同一 RSVP ルータ上で起動させ、内部で COPS セッションを確立させた。この RSVP ルータを経由するようホスト 1, 2 の間で RSVP セッションを確立させ RSVP メッセージ (PATH:ホスト 1→2, RESV:ホスト 2→1) 転送に要する時間を測定した。これをホスト 3, 4 間の RSVP セッション数や PDP 内のポリシ条件数を変えて実施した。但し, RSVP セッションに対応するデータトラフィックは送出していない。また, 全てのノードは NTP によって時刻同期させた。

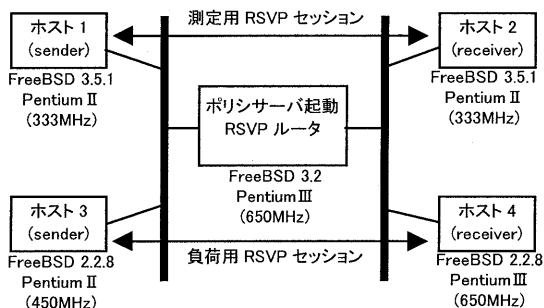


図 2. 実験構成

3.2. 実験結果と考察

図 3 はホスト 3, 4 間で予め確立しておいた RSVP セッション数に対する新規 RSVP メッセージの転送時間についてポリシ問合せを行う場合 (△/▲) と行わない場合 (○/●) とで測定したものである。但し, PDP では「全ての RSVP メッセージ処理を許可する」という最小限のポリシルールに留めた。図 3 より, ポリシ問い合わせの有無による RSVP メッセージ処理時間の差は確立済み RSVP セッション数には依存しないことが分かる。これは, PDP への問い合わせは RSVP メッセージが新規に作成された場合もしくはセッション内容に変更が生じた場合にのみ生じ, 確立済み RSVP セッションの定期的なリフレッシュ用 PATH/RESV メッセージに対しては生じないためである。

次に, ポリシ判断機能が参照するポリシルール内の条件数を変えて測定を実施した。ポリシ条件ファイルの記述はファイル先頭より測定用 RSVP メッセージに適合しないダミーの条件文で埋め, 最後の行に同メッセージに該当する条件文を挿入した。測定結果を図 4 に示す。ポリシ条件数に比例してメッセージ処理時間が增大しているのがわかる。グラフの傾きが PATH, RESV メッセージとも等しいのは両メッセージに対して同じポリシルールを適用させたこと,

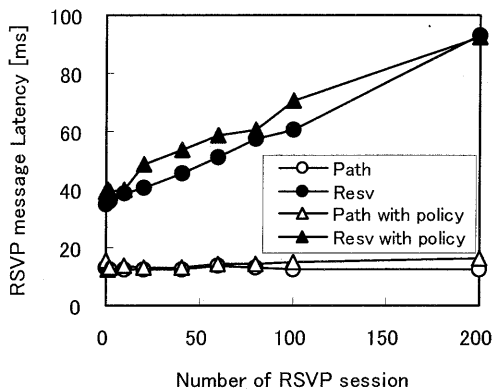


図 3. RSVP セッション数と転送時間

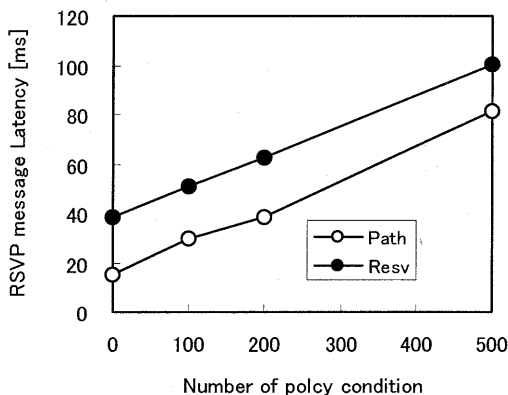


図 4. ポリシ条件数と転送時間

また RSVP メッセージ転送に要するポリシ問合せ回数を RATH, RESV とともに 2 回 (in, out) としたためである。従って, 今回使用した PC (CPU:Pentium III 650MHz, memory:128MB) では 1 回のポリシ問合せについてポリシ条件 1 つを検証するのに要する時間が約 0.06ms であったといえる。適用するポリシルールの条件数が 100 程度のネットワークであれば RSVP メッセージ転送のオーバーヘッドは RSVP プロセス部の処理が支配的であるため本システムは十分適用可能であるといえる。

4. おわりに

RSVP 網におけるポリシベース簡易 QoS 管理システムの実装について報告し, RSVP メッセージ転送時間をもとにシステムの性能評価を行った。スケーラビリティを考慮したポリシ管理ドメインの階層化やディレクトリサーバを導入した場合との性能比較などについては今後の課題である。

日頃御指導頂く KDD 研究所秋葉所長に感謝いたします。

参考文献

- [1] Y. Ito. et al, "Bandwidth-guaranteed IP tunneling router with RSVP," IEEE IPCCC'98, pp.291-297, Feb. 1998.
- [2] IETF RFC 2748, "The COPS (Common Open Policy Service) Protocol," Jan. 2000.