マルチメディア通信と分散処理 103－17
（2001．6．8）

NATゲートウェイを通過するフローの網内での識別手法の提案

横山輝明[†1] 森島直人[†1] 小川晃通[†2]
宇夫陽次朗[†3] 宇多仁[†3]
江崎浩[†4] 山口英[†1]

現在のインターネットでは、NATをもちいたサービス網構築が広く行なわれている。しかし、NATを通じた通信ではフローの識別子であるアドレスやポート番号が変換され、その前後での一連のフローを異なったフローと識別する。その結果、通信の始点、終点のアドレスを必要とするアプリケーションや、Diffservなどのフローの識別を必要とするサービスの提供が困難となる場合がある。本稿では、NATの変換表をアプリケーションやサービスに伝達し、一貫したフロー検出を可能にするための機構を提案する。さらにその手法に基づくシステムを実装し実験ネットワークで運用することにより、提案手法の有効性について評価を行った。この実験結果より本手法の有効性を確認した。

Proposal of the mechanism to identify the End-to-End flow over NAT gateways

Teruaki Yokoyama,[†1] Naoto Morishima,[†1] Akimichi Ogawa,[†2]
Yojiro UO,[†3] Satoshi Uda,[†3] Hiroshi Esaki[†4]
and Suguru Yamaguchi[†1]

Using the NAT has become quite popular in the Internet, especially for commercial ISP's environment for reducing its consumption of global IP address assignments. In order to identify an identical flow over NAT gateways, we proposed the new filtering architecture and scheme effectively working with the applications and services that require the end point information of the flow. In this article, we show our design and evaluation of its trial implementation of the mechanism.

1. はじめに

インターネットの拡大によりIPv4アドレスの消費が急激に進み、その枯渇問題が指摘されるようになった。この問題への短期的な解決手段として、NAT(Network Address Translation) [1] が提案され、多くのISP等で利用されている。NATでは、プライベートアドレス [2] をもつプライベートネットワークを構築し、インターネットとの境界にIPv4アドレスやTCP/UDPポート番号の変換器が設置され、すべての通信は変換器を通して行われる。変換器は通過するIPデータグラムのプライベートアドレスとグローバルアドレス、および、TCP/UDPポート番号を適切に変換し、プライベートネットワーク上のホストに対してインターネットへの接続性を透過的に提供する。

インターネット上を流れるフローを識別したい場合、そのフローが使用するトランスポートプロトコル、送信ホストと受信ホストにおける通信主体 (peer entities) が用いるIPアドレス、および、ポート番号を用いることで識別することが可能である。ここでは、この5つ組の情報を識別情報と呼ぶ。変換器を通過するフローに注目すると、変換器の前後ではプライベートネットワーク上のホストを指定するIPアドレスおよびポート番号は異なる。このため同一のフローであっても、プライベートネットワーク内と、プライベートネットワーク

†1 奈良先端科学技術大学院大学
Graduate School of Information Science, Nara Institute of Science and Technology
†2 慶應義塾大学 政策メディア研究科
Graduate School of Media and Governance, Keio University
†3 北陸先端科学技術大学院大学 情報科学研究科
School of Information Science, Japan Advanced Institute of Science and Technology
†4 東京大学大学院 情報理工系研究科
Graduate School of Information Science and Technology, The University of Tokyo

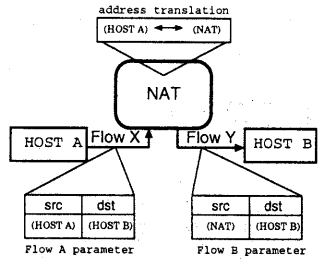

図 1 NAT の動作

外での識別情報は異なる。ある特定のサービス☆では、この識別情報が上位層、特に、アプリケーションで用いられているために、NAT を介したことにより正しい識別情報が与えられず、結果としてサービスが提供できない状況が発生している。このことから、変換器を通過しても同一フローを同一と識別し、その結果をアプリケーションが利用できる手法が必要となる。この手法を、ここでは「フローの一貫した識別」手法と定義する。

本稿では、NAT の介在する環境において通信の識別を必要とするサービスの提供を可能とするために、変換器が中継する通信において一貫した識別を可能とする手法を提案する。提案手法では、変換器の持つプライベートアドレスとグローバルアドレス、および、TCP/UDP ポート番号の変換表をサービスに通知し、サービス側において識別ルールの変換を行うことで通信の一貫した識別を可能にする。また、本手法を実現するためのプロトコルおよびサーバの設計と実装を行った。さらに、通信の識別を必要とするサービスの例として Diffserv を挙げ、Diffserv をもちいたネットワークに適用し、提案手法の有効性を確認した。

2. NAT が介在した通信での問題

NAT[1] では、プライベートネットワークとインターネットの境界に変換器を設置することによって、それらの間の透過的な通信を提供する。IPv4 データグラムが変換器を通過する際、変換器は内部に蓄積されたプライベートアドレスとグローバルアドレス、および、TCP/UDP ポート番号の変換表にしたがって IP ヘッダを書き換える。また、IP データグラムに対応するエントリが変換表に存在しない場合、新たなアドレスの対応を動的に生成して変換表に蓄積する。この変換エントリの動的な生成は、プライベートネットワーク内部からインターネットへの通信の場合にのみ行われる。したがって、NAT の透過的な通信の提供は、セッションがプライベートネットワークからインターネットへ向けて開始された場合のみである。動的に生成されたエントリは、対応する通信の終了から一定時間が経過すると変換表から削除される。

一方、高付加価値ネットワークを実現するためのサービスやアプリケーションには、通信主体間を流れるデータの識別を必要とするものがある。この通信主体間のデータの流れをフローと呼ぶ。これらのサービスやアプリケーションは、一般に IP アドレスや TCP/UDP ポート番号といった IP データグラムに含まれる通信主体を表すパラメータの集合によってフローを識別する。このパラメータの集合を、フローの識別情報と呼ぶ。一般に識別情報として、フローが使用するトランスポートプロトコル、通信主体が用いる IP アドレスとポート番号の 5 つの情報を用いることが多い。

しかし、NAT が介在した環境では通信主体間でフローの一貫した識別ができない。これは変換器が IPv4 アドレスや TCP/UDP ポート番号を変換するために、フローを表す識別情報としてアドレスやポート番号を用いた場合には、変換器の前後で、通信主体間での同一フローが異なるフローとして識別されるためである (図 1 参照)。このために、NAT が介在した環境ではフローの識別を必要とするサービスの提供が困難となる。特に、次世代インターネットにおける中核技術のひとつと考えられている Diffserv を用いたサービスの場合、NAT の利用はフローの識別を困難にする。このことから、NAT が通信に介在しても、一貫したフロー識別を可能とする手法の実現が強く望まれている。

3. 解決手法の提案と実装

本章では、これまで述べてきた NAT の介在する環境でのフロー識別問題に対する解決手法を提案する。さらに、提案手法の実現に必要なコンポーネントの設計と実装、および、各コンポーネント間の協調について議論する。

3.1 解決手法の提案

変換器によって変換されるフローを観測した場合、インターネット側とプライベートネットワーク側では、フローの識別情報を構成するパラメータの値が異なる。本研究では、変換器を通過するフローについて、プライベートネットワーク内での識別情報を構成するパラメータの値を同定し、インターネット側に存在する通信主体から利用できるような機構を提案する。これにより、インターネット側の通信主体では、インターネット側での識別情報と、プライベートネットワーク側での識別情報のマッピングを生成することができる。これにより、プライベートネットワーク側の通信主体から与えられた識別情報に基づいて何らかのサービスを与え

☆ 例えば Diffserv を適用したサービス。

る場合、得られたマッピングを適用することで、同一フローの識別が可能となる。さらに、この機構を利用することで、インターネット側でフローを中継するノードでのフィルタリングなどにも応用することができる。

この手法を実際のサービスで利用するためには、(1) 識別情報を構成するパラメータの集合の定義、(2) 変換器での、NAT 機構が蓄積するアドレスおよびポート番号の変換表の取得、(3) インターネット側の通信主体に対する変換表の通知、(4) インターネット側の通信主体における識別情報のマッピングの生成、(5) 識別情報のマッピングを利用するサービスへの適用、の 5 つの対応が必要になる。

3.2 識別情報の定義

本研究では IPv4 環境下での NAT を対象とするため、使用するトランスポートプロトコルの種別、通信主体の IP アドレス、および、ポート番号の 5 つのパラメータを識別情報を構成するパラメータの集合とする。

3.3 変換表通知サーバとプロトコルの設計

前節で提案した手法を実現するために、(2) および (3) を実現するサーバを変換器上で実現する、このサーバを変換表通知サーバと呼び、インターネット側の通信主体に対して通知するためのプロトコルを変換表通知プロトコルと呼ぶ。次節以降では、サーバとプロトコルの機能の実現方法について議論する。

(A) NAT が蓄積する変換表の取得

変換表は変換器を通過するセッションの開始と終了にともなって増減する。このため、変換表の取得機構はその増減に即座に追随することが求められる。変換表取得機構を駆動するイベントには次の 2 種類がある。

- ポーリングによるタイムアウトイベント
- 変換表を更新した変換器からの割り込み

前者では、事前に定められた一定時間間隔で変換表通知サーバがポーリングによって変換表を取得する。一方、後者では、変換器が変換表更新を割り込みによって変換表通知サーバへ通知する。変換表の増減に即座に追随するためには割り込みによる取得が望ましい。しかし、一般に NAT 機構の実装はカーネル空間内で行われており、変換表通知サーバをユーザ空間に実装した場合には、カーネル内スレッドからユーザ空間に対する割り込み機構を実装しなければならない。

(B) 変換表のサービスへの通知

変換表通知サーバは取得した変換表をサービスに対して通知する。この通知には 2 通りの方法が考えられる。

- 変換表の全てを送信する
  変換器から取得した全ての変換エントリを毎回送信する方式。この場合、変換表通知サーバでは変換表の送信前処理を行なう必要は無い。しかし、送信される変換表は、前回送信した変換表とエントリが重複する可能性があるため、サービス側におけるエントリの重複チェックが必要となり、実装が煩雑となる。また、すべての毎回エントリを送信するため、サービス側との通信での実効帯域が狭い場合には通信帯域の圧迫が問題となる。
- 変換表の差分を送信する
  前回取得した変換表と今回取得した変換表との差分を得て、増減の発生したエントリのみを通知する方式である。この場合、変換表通知サーバで直前の変換表を保持し、送信前に比較処理によって差分を生成する必要がある。サービス側におけるエントリの処理は各エントリに付加された指示に従う。

ここでは、受信側のサービスの負荷やネットワークへの影響を最小限におさえるため、変換表の差分を送信する手法を採用する。

サービス側に通知する情報には、先に定義した識別情報を構成するパラメータの集合を含む必要がある。このために、通知パケットは、変換表の差分に応じて図 2 に示す通知エントリを複数含む。通知エントリは変換表の 1 エントリに対応する。ただし、この構造はネットワーク層プロトコルとして IPv4 を仮定しており、IPv6 に対応するためには構造の変更が必要となる。

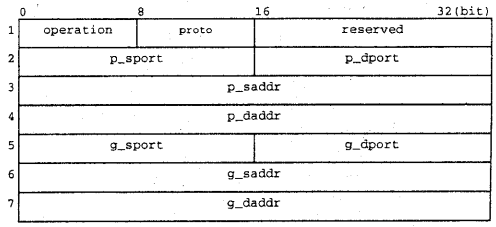

図 2 通知エントリの構造

各フィールドは以下の情報から構成される。

- **operation**
  通知エントリが、追加なのか削除なのかを表す。
- **family**
  上位層プロトコル ID が格納される。
- **reserved**
  将来のための拡張用フィールドで現在は未使用であり、0 で初期化される。
- **p_saddr, p_sport, p_daddr,p_dport**
  プライベートネットワーク側で観測した場合のフローのパラメータ値。それぞれ、送信元アドレス、送信元ポート、受信先アドレス、受信先ポート。
- **g_saddr, g_sport, g_daddr,g_dport**
  インターネット側で観測した場合のフローのパラメータ値。それぞれ、送信元アドレス、送信元ポート、受信先アドレス、受信先ポート。

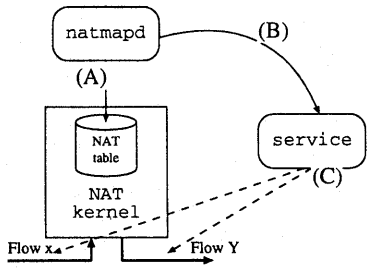

図 3　提案手法を実現する機構

通知パケットのサイズは 28 octet である。通知パケットの個数を減らして通信帯域を削減するため、送信すべきエントリが複数ある場合でもひとつの通知パケットに含めることができるようにすることとする。

(C) サービス側での対応

通知される変換表から、インターネット側のサービスでは、インターネット側での識別情報と、プライベートネットワーク側での識別情報のマッピングを生成する。このマッピングを利用することで、フローの一貫した識別が可能になる。インターネット側のサービスでは、変換表通知サーバから与えられる情報を用いるための機構を追加する必要がある。

以上で示した変換表通知機構とそれに関連するコンポーネントの協調関係を図 3 に示す。図中のラベルは、上記の処理に対応している。

3.4　変換表通知サーバの実装

前節の設計にしたがって変換表通知サーバ **natmapd** を実装した。変換器は FreeBSD 3.5-RELEASE および NetBSD 1.5 のカーネル内に実装された NAT を対象とした。これらの変換器の実装では、変換表更新を通知するための割り込み機構が用意されていない。このため、ここでは提案手法の有効性を確認することを目標とし、実装が簡便に実現できるポーリングによる手法を実装した。ポーリング手法を用いた場合、変換表更新を確実に処理に反映するために、ポーリング間隔を適切な値に設定する必要がある。ここでは、変換表更新をできるかぎり正確に検知するため、変換表取得後の処理にかかる時間のみをポーリングの間隔とした。つまり、変換表通知サーバは、常にカーネル内の変換表を走査するような実装となっている。しかし、変換表通知サーバはユーザ空間に実装されるために、カーネル内での NAT 本来の処理、および、IP データグラム転送機能の性能を損なうことはない。変換表へのアクセスはカーネルが提供する kmem インターフェイスを利用している。

変換表通知サーバでは、カーネル内の変換器から取得した内部情報により、サービスへ通知するアドレス

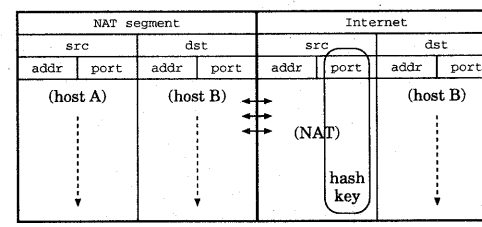

図 4　natmapd の生成するアドレス変換表

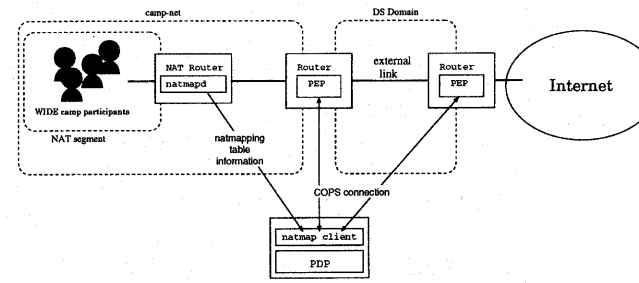

図 5　実験ネットワーク

変換表 (図 4) を生成する。この変換表は変換器の前後におけるフローのパラメータの変化を表わし、変換表の 1 行がアドレス変換ルールの 1 エントリに対応する。

先に述べたように、この変換表と、前回取得した変換表との差分を計算し、差分のみをサービスに対して送信する。

差分計算ではエントリの検索処理が多数発生し、変換表通知にかかる処理時間が増大することにもなる。変換器の機能を考慮すると、変換表のインターネット側の src の範囲は、変換器の持つアドレスやポート番号となる。そこで、この変換表はインターネット側の src port の値 (0 から 65535) をキーとするハッシュ方式のデータ構造を持ってメモリに格納する。ハッシュを使用することで、差分計算のエントリ検索の高速化を実現している。

サービスはこの変換表を取得し、変換器の通過前のフローの識別情報と、通過後のフローの識別情報のマッピングを生成する機構を持つ。

4.　提案手法の運用実験

Diffserv[3] は、フローの識別を必要とするサービスである。本稿での提案手法の Diffserv への応用実験を行なった。

4.1　実験環境

WIDE プロジェクト [4] は、産官学共同のインターネット技術研究開発コンソーシアムである。WIDE プロジェクトでは、毎年春秋に合宿形式の研究会を行なっている。この研究会では、参加者に対してインターネッ

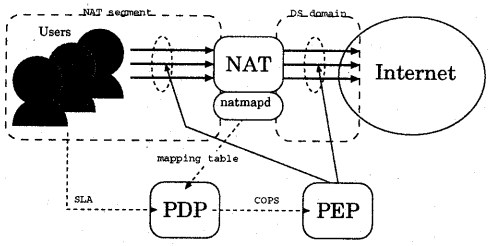

図 6　フロー識別の動作概要

トへの接続性と実験環境の提供のために、camp-net と呼ばれるネットワークが、研究会開催場所に構築され運用される。本研究で開発したシステムの有効性を確認するために、camp-net(図 5) を利用して、運用実験を実施した。2001 年春の WIDE 合宿は、2001 年 3 月 5 日から 7 日にかけて開催され、270 名が参加した。

WIDE 合宿への参加者に対するインターネットへの接続性は、camp-net の NAT セグメントやグローバルセグメントに接続することによって得られる。

4.2　フローの識別を必要とするサービス

Diffserv はインターネットにおける通信品質の保証を行なう機構である。Diffserv によるサービスの提供を受けるユーザは、ネットワークプロバイダと SLA (Service License Agreement) と呼ばれる通信品質を保証する契約を結ぶ。SLA には、通信の帯域、遅延、遅延の揺らぎなどが規定される。

同一ポリシによって運営される Diffserv ネットワークによって、DS ドメインが形成される。DS ドメインへ流入するフローは、DS ドメイン入口ノードで SLA に基づいて分類され、その SLA を表す DSCP が付加される。DS ドメインの内部では、フローは DSCP 毎に集約され、集約されたフローは 1 ホップ毎に DSCP に応じた優先制御が行なわれる。この集約を BA (Behavior Aggregate)、1 ホップ毎の優先制御を PHB (Per Hop Behavior) と呼ぶ。

Diffserv の特徴は、個々のフローそれぞれに対して優先処理を行なうのではなく、DSCP によって集約されたフロー全体に対して優先処理を行なうことである。Diffserv の機構では、DSCP の付加、PHB の切り換えのそれぞれの処理でフローの識別を必要とする。

Diffserv ではアドミッション制御の機構は定義されていない。今回の実験ではアドミッション制御として COPS (Common Open Policy Service)[5] プロトコルを利用する。COPS は、ポリシの決定を行なうノード PDP (Policy Decision Point) と、ポリシを反映するノード PEP (Policy Enforcement Point) によって構成される。本実験で使用した Diffserv によるサービス提供の概要を図 6 に示す。

PDP はユーザからの予約情報を保持し、SLA を履行すべきフローを識別するためのフィルタスペックを生成し、PEP に対して通知する。PEP は、このフィルタスペックによって特定されるフローを優先制御する。

運用実験を行ったネットワークでは、NAT セグメントから Diffserv を利用するユーザが存在し、NAT セグメント上のユーザは SLA を PDP へ送信する。このときの SLA によって生成されるフィルタスペックによって識別されるのは、NAT セグメント内のフローである。PDP, PEP はインターネット側に位置し、このフィルタスペックからは実際に優先制御を行なうべきフローを特定することができない。

運用実験では、フローを特定するための手法として、本研究が提案/実装した **natmapd** を利用する手法を適用した。これより、PDP, PEP が NAT セグメントから提供されるフィルタスペックを用いても、インターネット側でのフローを検出/同定し、これにより Diffserv に基づいた優先制御を実現できる。この実験では、実際に WIDE 合宿参加者に Diffserv を利用してもらい、参加者によって生成されるフローを対象として、本機構が正しく機能するかどうかを確認した。

4.3　実験結果

上記の機構を用いて、WIDE 合宿開催期間の 3 日間で連続した実験を行ない、NAT セグメント上のユーザからの SLA を処理した。PDP は本手法によって、NAT の前後でフローを一貫して識別することが可能となり、NAT を通過したフローのうち、SLA を適用するものを識別するフィルタスペックを構成することができた。このフィルタスペックを使うことで、NAT セグメント内からの SLA を履行することができ、Diffserv のサービスを NAT セグメント内のユーザに対して提供することを確認した。これにより、本機構が正しく機能していることを確認することができた。

5.　考　　察

NAT の内部情報を通知することで、通信の始点から終点までで一貫したフローの識別を行なうことができる。本手法の有効性は、運用実験を通して確認できた。しかし、この実験によりいくつかの問題が明らかとなった。

5.1　設計上の問題

通知フィルタの概念の導入

natmapd は変換器の変換表全体について差分をサービス側に通知する。しかしながら、あるサービスにとっては、特定のフローについてのみ識別情報のマッピングのみを生成すれば十分である。現在の **natmapd** が実装では、あるエントリがサービスにとって要、不要かを判断することなくエントリを通知する。このため不

要なエントリについての情報が通知され、さらに無駄なマッピング生成が行われてしまう。

この問題を解決するためには、サービス側から変換表通知サーバに対して、エントリの要、不要を通知する機構が実装されればよい。本機構については現在実装を進めており、次バージョンで対応を考えている。

より一般的な変換器への対応

今回の設計では、IPv4 環境での NAT 機構に対応する設計を行った。しかし、現在のインターネット技術では、通信経路上で通信主体を表すアドレスの変換 (NAT) だけではなく、その他の情報を変換するサービスが考えられる。具体的には、IPv4/IPv6 プロトコル変換などが考えられる。このようなことから、識別情報を構成するパラメータは、単に IP アドレス、ポート番号、トランスポート層プロトコルのみに限定されるものではなく、より多くのパラメータの取り扱いが必要となる。このようなことから、変換表通知機構、および、通知プロトコルをさまざまな種類のパラメータに対応できる構造を導入することが望まれる。さらに、サービス側からどのパラメータが必要となるのかを通知し、変換表通知サーバが指定された情報を与える機構を実現することが望ましい。

5.2 実装上の問題

NAT の内部情報取得のタイミング

今回実装した **natmapd** では内部情報をポーリング方式で取得したため、**natmapd** の内部情報を取得する周期よりも短いタイミングで発生、終了するフローがあった場合、そのフローの情報は取得できない。このようなフローを **natmapd** クライアント (インターネット側サービス) は検知することができない。

この問題の解決には、割り込みによる通知駆動方式を用いる必要がある。割り込み方式では、NAT が変換表の更新を割り込みによって変換表通知サーバへ通知する。この割り込みの機構をカーネル内に実装し、より適切な通知が実現できるようにする必要がある。

SDK の提供

本研究で開発した変換表通知サーバを利用するサービスでは、識別情報のマッピングの生成などの処理が必要となる。これらの処理を簡潔に記述し、サービスを本サーバに対応するための改造作業を簡単に行うために、変換表通知サーバを利用するための API を定義し、さらに、その処理を提供するライブラリを SDK (Software Development Kit) として提供することが望まれる。

6. おわりに

IPv4 アドレスの枯渇問題が指摘されるようになり、この問題への短期的な解決方法として NAT の利用が提案され、インターネット環境で広く使われている。しかし NAT が行うアドレス変換によって、フローを識別する必要のあるサービスを提供することが困難となってしまった。この問題を解決するために、本研究では、NAT を介したフローの一貫した識別を実現する手法を提案した。この手法では、インターネット側のサービスに対して、プライベートネットワーク側での識別情報を通知し、インターネット側で稼働するサービスが識別情報のマッピングを生成することを可能としている。この機構を利用することで、サービスが NAT の存在に関わらず、サービスを提供する基盤を実現することができる。

本研究では、提案手法を実装し、運用実験を通してその有効性を確認した。現在の実装は、提案手法を確認するために短時間、かつ、簡便に実装できる手法を採用しており、実運用を考えた場合には、改良の余地が数多く存在する。さらに、NAT 以外のプロトコル変換を行う通信方式、特に IPv4/IPv6 プロトコル変換に対応するための機構拡張が望まれている。今後、実装したシステムを改良/拡張し、拡張性に富み、実運用に耐えられるシステムを提供する予定である。

参考文献

1) P. Srisuresh and K. Egevang. *Traditional IP Network Address Translator (Traditional NAT)*, January 2001. RFC 3022.
2) Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear. *Address Allocation for Private Internets*, February 1996. RFC 1918.
3) S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. *An Architecture for Differentiated Service*, December 1998. RFC 2475.
4) WIDE Project. http://www.wide.ad.jp/.
5) J.Boyle, R.Cohen, D.Durham, S.Herzog, R.Rajan, and A. Sastry. *The COPS (Common Open Policy Service) Protocol*, January 2000. RFC 2748.