Policy Based Architecture for QoS over Differentiated Services Network

Priyadarsi Nanda
Faculty of Information Technology,
University of Technology,
Sydney, Australia.
E-mail: pnanda@it.uts.edu.au

Andrew James Simmonds
Faculty of Information Technology,
University of Technology,
Sydney, Australia.
E-mail: simmonds@it.uts.edu.au

ABSTRACT

Current Internet architecture is based on the Best Effort (BE) model, where packets can be dropped indiscriminately in the event of congestion. This architecture attempts to deliver all traffic as soon as possible within the limits of its abilities, but without any guarantees about throughput, delay and packet loss etc. Though such a model works well for certain traditional applications such as FTP, E-mail and less QoS constrained applications, it can be intolerable for newly emerged real-time, multimedia applications such as Internet Telephony, Video-Conferencing and Video on-Demand. This paper is based on the on-going research activities being carried out by various researchers in the area of QoS and proposes a Policy Based Network (PBN) architecture for the Differentiated Services (Diff-serv) Network. Policy Based Networking received much attention recently as the devices within the networks can be implemented with greater control. Our proposed architecture is based on the functionalities defined within the existing IETF/DMTF Policy architecture, with an objective to achieve QoS through proper Resource management techniques.

Keywords
QoS, Diff-serv, Policy Based Network (PBN), Resource allocation, Resource management, COPS.

1 INTRODUCTION

The study of QoS related issues over the Internet for real-time data transfer mechanisms in recent years has resulted in two different architectures defined by the Internet Engineering Task Force (IETF): Integrated services (Int-Serv) network and Differentiated services (Diff-serv) network. Int-serv makes use of the Resource Reservation Protocol (RSVP) for signaling and is aimed at achieving the desired end-to-end QoS on a per flow basis. Such a mechanism can certainly achieve QoS with fine granularity on an end-to-end basis, but unfortunately it is not scalable to the Internet due to the problem of managing millions of traffic flows within the core of the network. Diff-serv is scalable because it deals with aggregated traffic flows on a per class basis rather than individual flows, but because of that Diff-serv offers less granularity than Int-serv, and on its own does not guarantee end-to-end QoS. Diff-serv involves the process of packet classification and packet marking (with a Diff-serv Code Point DSCP) at the edge of an Autonomous System (AS) or domain, and defining appropriate Per Hop Behaviors (PHBs) in all routers for relevant applications. An edge router may be differentiated into providing both ingress and egress routing function, depending on the direction of the traffic. A Diff-serv enabled router will inspect an incoming packet's DSCP to determine the appropriate PHB to be applied to that packet, before doing its normal function of switching the packet. Diff-serv and its currently evolved standards within the body of IETF rely mostly on administrative control of bandwidth, delay or packet dropping preferences although the standard lacks any proper definition on which signaling methods to be used. Such administrative control can be enforced through a policy, where a policy is defined as a set of rules which when invoked produces some action. The set of rules represent some high level directives which may contain information such as pricing, access rules, time or day of use, etc.

At the user level, the set of rules may define a few high-level directives enforced between the user network and its immediate service provider, e.g. ISP, through a Service Level Agreement (SLA). These high-level policies need to be translated and validated into low-level policies which are dependent on the specific technologies such as: Diff-serv, Int-serv, MPLS, etc. In this paper we have focused specifically on Diff-serv networks, as our goal is to achieve overall scalability through the proposed architecture. Policies may be specified for controlling edge nodes, classifying traffic flows, and enforcing decisions related to

admission control with or without addressing end-to-end resource management. One of the critical issues for designing a policy capable resource management component is to relate the parameters at the device level that will result in different allocation of resources in terms of business decisions. In this paper we investigated some results of current research on next generation Internet architectures [1, 3, 7, 13] and propose Policy Based Network architecture particularly in relation to Diff-serv Network. We believe this can achieve both scalability and finer-grained end-to-end QoS (through the use of signaling protocols). Our proposed QoS architecture is based on a three-tier policy model, where each level is able to manage a set of functions and results in certain policy actions enforced between the devices at different levels. In this approach we can achieve our objectives, as long as certain policy actions are defined and enforced for a set of devices in each level of our architecture. Our work in this paper is presented through the following sections.

Section 2 in this paper presents a review of recent work on policy based network architecture. In section 3 we give our proposed model and highlight the differences to the models reviewed in section 2 and present a case study for a corporate level network and its service provider's architecture based on our proposed frame work. Finally, section 4 presents conclusion and proposals for future work.

2 RELATED WORK

There has been a considerable effort both in the industry and research community in recent years investigating and developing new technologies to provide QoS over IP based networks. The IETF tried to resolve the issue by proposing Int-serv architecture (RFC1633) With RSVP signaling. One of the major drawbacks of such an approach was the scalability issue. Later, the IETF defined a new model called Diff-serv (RFC 2475), within which it defined a set of traffic classes each of which is designated to serve applications with similar QoS demands. Hence the Diff-serv model was more focused towards resolving the issue of a scalable architecture for the Internet by treating each class as an aggregated flow. The Diff-serv model achieves scalability by pushing the complexity of traffic classification, policing and shaping from the core of the network towards the edge/border; whilst core routers inside the network only have the task of prioritized routing by looking into the IP packet header information to determine the DSCP and carrying out the required PHB for the flows. Hence, the core routers need to know only the traffic classes and their corresponding PHBs. One major problem in the Diff-serv model which has yet to be resolved is signaling between users and the edge routers. One solution is to include a logical entity called a Bandwidth Broker (BB) [1,7], which can be implemented in each network domain (or there may be several in each domain) and which will handle signaling, traffic management and resource management. A few of these functions are handled by the Policy Decision Point (PDP), defined within the IETF Policy workgroup and discussed below. We envisaged the need for such an entity in our previous work [2, 10, 11] for signaling, but we concentrated on using the BB for resource allocation and achieving end-to-end QoS rather than being part of our proposed PBN.

Handling the resource management issue is an important aspect of guaranteeing QoS to specific applications (such as VOIP and multimedia traffic), but the current Internet architecture was never designed to handle such an issue. Hence the need for better management of time constrained applications and the need for an improved architecture is an important aspect of current research in the Internet. Resource allocation can be either policy based or constraint based, or both. Essentially constraint based allocation has a null policy of admit everything. The constraint based allocation succeeds if the Call Admission Control (CAC) process at the edge router determines resources are free. E.g., if the user is allocated x Mbps of EF traffic, anything above this is simply rejected when the Edge router applies its CAC. A pure policy based allocation process could allocate unavailable resources if the real state of the network is out of step with the policy resource database. Hence there needs to be some feedback mechanism to report the real state of the network, either to update the policy resource database, or to update the CAC process in the edge router (as one of us proposes in [8]). In this paper, our proposal uses a policy based resource allocation and assumes the CAC process will not admit a new call if resources are not available.

Policy based management has been the subject of extensive research over the last decade and currently there are two working groups within the IETF which have considered policy based network architectures: the Resource Allocation Protocol (RAP), and the policy framework working groups [13, 15]. In a Diff-serv network, policies need to be specified for controlling edge nodes by enforcing decisions related to admission control, classifying traffic flows and assigning/checking DSCP to the packets within a flow. Similarly policies related to individual PHB must be enforced by the Core routers. It may happen that more than one device within a network may be configured the same policy. In this case devices having the same policy requirements should be configured equally. E.g. all the core routers for a traffic flow may be configured for the same type of queuing disciplines within a network.

For this paper, we looked through several works and found many issues to be resolved in relation to deployment of QoS. These include: administrative criteria determining the number of applications or users allowed to have

preferential access to network resources, optimization of the network resources to meet the requirements of the contracted services, intra domain resource provisioning in an automated manner from an ISP's perspective, management of services using admission control policies and traffic engineering. Theses issues were covered in common by refs [3, 5, 9, 12, 13].

Policy plays an increasingly important role in QoS management, both within the corporate intranets as well as in the public Internet. IETF's RAP working group described a framework having two major components involved in the process of resource management. They are the Policy Enforcement Point (PEP) and the Policy Decision Point (PDP). The PEP and PDP could be two different functions in the same device, or even the same process running on the device. We considered the PEP process being embedded within the Edge router itself while PDP process runs separately on a Server. PEP enforces the policy actions on the networking devices such as the edge and core routers while PDP decides on the appropriate actions for specific user/applications and delegates those actions to the PEP. Policy decisions are always made at the PDP (e.g. a Bandwidth Broker). Also the IETF has defined the Common Open Policy Service (COPS) protocol for signaling in the context of the RAP working group as a means to support policy control [14, 15].

There are two main models currently supported by the COPS protocol: the Outsourcing model and the Provisioning model. Under the Outsourcing model, user requests concerning flows are noticed by the PEP at the first instance and then the PEP passes either the complete request or part of it to the PDP, and finally the PDP takes the decision on whether to allow the flow into the network or not. On the other hand, under the provisioning model, the PDP proactively configures the resource handling mechanisms in the PEP, i.e. network elements are preconfigured, based on policy, prior to user requests. While the Outsourcing model is reactive and utilizes network resources more intelligently, the Provisioning model is proactive and there is no set-up delay. Our proposed architecture takes both models into consideration and applies either or both of them depending on requirements. In addition, we also envisage a third case we call the Direct-request model, where in a lightly loaded domain there is no need to reserve resources between the user and the egress router, hence in this case the user would signal the PDP directly that it requires inter-domain QoS resources. If the request is within policy, the PDP grants the request and informs the edge router to allocate resources. The edge router performs CAC (may be better referred as Call Exit Control) and if resources are available the flow is accepted.

As mentioned earlier in this section about different works carried out recently on policy based network architecture, most of them are only applicable within a single domain. It is our argument that this is unrealistic: Internet traffic may travel across several domains from the source till it reaches the destination network. We refer to the networks run by Internet Service Providers (ISPs) or Autonomous Systems (AS) as domains and each domain is configured to have its own policies and rules as a result of which policy conflict is an obvious scenario between different domains. Our aim is to provide a unified policy structure which can be easily adapted by these domains without much conflict in sharing resources. When we say unified, our proposed architecture is more scalable and is built upon a three-tier model and is different to the architecture [1]. We present our proposed policy based architecture in section 3 and define the function of each level and their necessity to support our architecture in a scalable way.

3 PROPOSED ARCHITECTURE

The proposed architecture given in this paper is based on a three-tier service module; each level being able to manage a set of functions for that level only. Communication between each level can be achieved through a single or a series of signaling protocols and our aim is to achieve scalability for the proposed architecture. Figure 1 below outlines our proposed Policy frame work for the architecture.

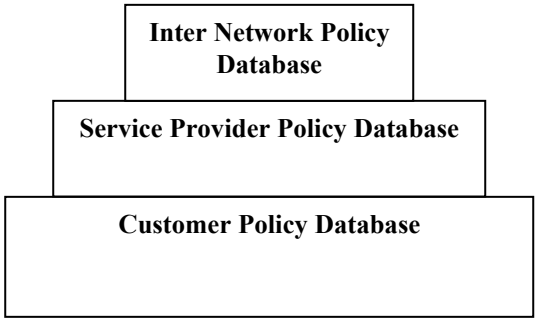

Figure 1: Policy frame work

This can be considered an extension of the two-tier architecture of [1], in which the top tier was bilateral agreements between neighboring domains, and the lower tier was essentially a hidden layer covering whatever a service provider did within their own domain. In our proposed architecture each level stores policies within a repository which may either be input by using a Policy Management Tool (PMT) or by the network administrator. The policies consist of rules which are derived from the SLAs between different domains. Once the policy rules are derived and stored in the repository, our framework uses a rule-based system for validating the policies and translating them to a set of configurable parameters intended for a specific/group of devices within the Diff-serv network.

Three major functions related to our proposed architecture/framework are presented in this section. They are Administrative control, Traffic Engineering and Resource management. As mentioned before, our particular example uses the Diff-serv network. Figure 2 below presents the set of actions that may be carried out at different layers in our architecture having layer 1 in the customer network and layer 2 and 3 are within the ISP.

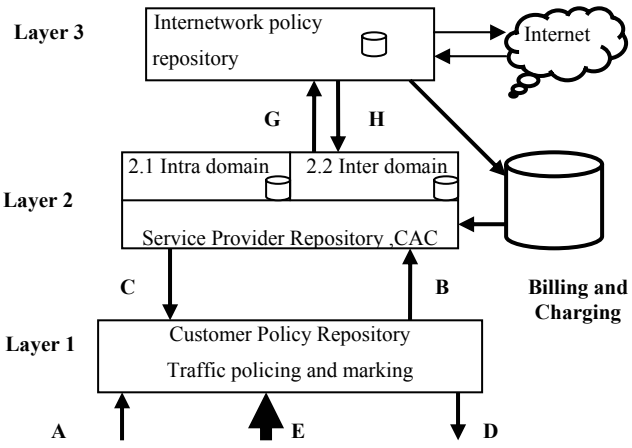

Figure 2: Layered communication architecture

As shown in the figure, each layer maintains a repository to validate different events that are represented through arrows for a hybrid model supporting both provisioning and outsourcing models as discussed in section 2. Below we present a case study in support of our architecture highlighting different events followed by the subsections describing the three major functions of our proposed architecture.

3.1 CASE STUDY DESCRIBING A HYBRID MODEL

For ease of understanding and implementation, we considered a simple scenario of a corporate network, Figure 3, where the corporate network (Network-1) is connected to its ISP (Network-2). Also shown in the figure, is a hierarchy of controlling devices such as PDPs and PEPs for our proposed hybrid model.

Network-1 at the first instance forecasts traffic estimates from the users of its own network and requests a SLA with Network 2 consisting of equivalent resource requirements for different traffic classes. After this, Network-2 (PDP) configures its own devices including the Edge router (PEP) and the Core routers through a set of policy actions to be effective for subsequent traffic flows carrying network traffic from Network-1. Simultaneously, Network-1 also configures its own devices and applies CAC algorithm for the traffic generated within its own domain.

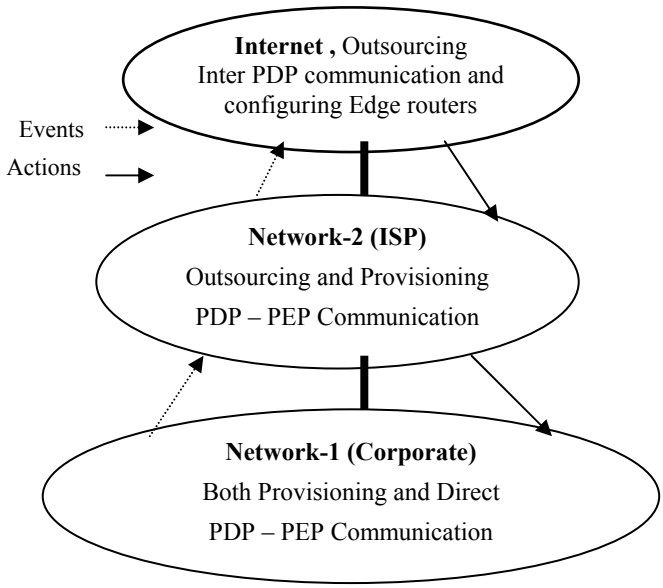

Figure 3: Hybrid model

Such actions are performed within the outsourcing model we discussed before. Now if Network-1 has few sub-networks within itself, the network administrator can pre-configure its sub-networks, resulting in a provisioning model within the whole architecture. It may so happen; those sub-networks are completely unaware about what is happening at the higher level.

As shown in Figure 2, the following series of actions need to be carried out through different levels of our architecture.

A: Initial forecasting and traffic estimation

B: SLA Request from customer policy base to ISP

C: SLA authentication and device configuration by the ISP

D: Configuring customer network device

E: Traffic flows from the user network to the service provider

F: Passing up of traffic profiles from Intra to Inter policy base in the service provider

G: Requesting for resources between ISPs

H: Successful Inter-domain resource reservation end-to-end and changing device configuration and pricing function for the user

In maintaining the policy bases at different levels and with proper signaling, overall management functionalities of our proposed architecture have been made scalable.

3.2 ADMINISTRATIVE CONTROL

Our main goal of designing policy based network architecture is to allow administrative objectives within the network to be translated into schemas for handling different packets with differential treatment. Such administrative objectives are expressed in terms of a Service Level Agreement (SLA) and are considered the first level of understanding between the user and their service provider. Our proposed architecture implements the administrative control functionalities through the 3-tier model as stated above having three distinct sets of components shown in figure 4 below:

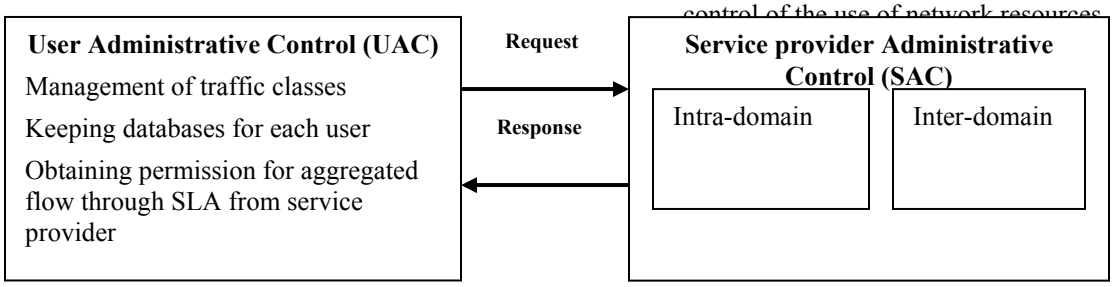

Figure 4: Two-tier Administrative control

Administrative control can be viewed under a two tier structure such as: User Administrative Control (UAC) and Service provider Administrative Control (SAC). There must be a good understanding between both UAC and SAC, otherwise SACs have to exert a number of controls every time they encounter traffic flows from the UAC. Further the functionality of the SAC may be decomposed into two sub-levels: intra-domain Administrative Control and inter-domain Administrative control. Hence the overall administrative functions in the Administrative control can be as shown in Figure 4. The UAC is confined to the user network. The user may be a corporation which has several physical networks connected within the same domain, or a single home user connected to a service provider. The UAC is responsible for managing its own user by maintaining their record on resources allocated to them. Depending on the demand for network resources from its own users, UAC sends a request to its service provider in accordance with its SLA and if granted, updates its user's record as well as allowing the user to send traffic.

On the other hand, the SAC has more responsibility than the UAC and divides its overall functionality of managing traffic flows into two sublevels. One of them, the intra-domain administrative control is responsible for managing traffic which originates or ends within its own service provider (or both, i.e. traffic which is wholly contained within the domain), while the inter-domain administrative control is responsible for resource negotiation, accounting, and other management functions to be enforced between itself and its neighbors. Such a distinction allows the overall functions of the service provider to be managed in a scalable way.

3.3 TRAFFIC ENGINEERING

In the proposed architecture, Traffic Engineering (TE) plays an important role in configuring the network elements, in monitoring and optimizing resource utilization and in forecasting resource requirements. This is because of the requirements to facilitate the maximum number of users consistent with their required QoS and to ensure tight control of the use of network resources.

Traffic Engineering is concerned with the performance optimization of operational networks [17].

The current Internet architecture performs packet routing based on the destination address using simple metrics such as hop count or delay count for routing decisions. Such an approach was never intended to optimize network resources within the networks. With destination based routing it is often difficult to manage load balancing, resulting in unbalanced distribution within the network.

Using the shortest path algorithm, it may so happen that if all the nodes use the same shortest path, the path may become easily congested and at that point a non-congested but nominally less optimal path may be better choice for forwarding the packets. In our architecture, we make the routing decision basing on the overall system objective and global view of the network. Our approach for resource optimization aims to achieve the following objectives:

- Improve utilization of resources amongst all the links
- Restrict overall delay experienced by packets to minimum

- Allow more users with existing resources
- Minimize packet loss and congestion

The first objective has been addressed by various works recently: In [6] the author introduces the concept of a 'pipe', where a pipe is defined as a logical path between two end points on the network having a predefined capacity. A pipe may be constructed for one to one, one to many and many to many ingress to egress routers. Also the author addresses dynamic resource management in a path for one to one ingress to egress router pairs within the Diff-serv network on a threshold based mechanism, but it is not clear how such a scheme can be applied to multi-domain resource management. In [14] a time-dependent IP Traffic Engineering and control system that operates at medium and long timescales has been presented. Such a system supported by traffic forecast can improve link utilization amongst links; though it still appears very complex at the moment for implementing. A different approach of path-oriented quota-based (PoQ) dynamic bandwidth allocation mechanism has been presented in [16], where a path is a 'pipe' between two edge routers in a single domain, which is made up of all the links between intermediate routers. In [16] the authors argue that such a scheme can improve the overall objectives of Traffic Engineering through the Bandwidth Broker. But, again, this does not address the issue of resource management in multiple domains.

Our proposal uses a policy based resource allocation process and pushes the decision on whether resources are available (i.e. constraints) to the edge router's CAC process. The proposed architecture supports the Provisioning model, while both the Outsourcing and the Direct-request model may be supported in addition when required.

3.4 RESOURCE MANAGEMENT

Resource management is considered one of the important aspects and is mainly built upon the layer-2 of our proposed architecture. The model looks at both Intra domain as well Inter domain resource management and maintains their related policies in the repository. A key aspect of allocating resources in the Diff-serv network is achieved through distributing bandwidth of the service provider in a dynamic way. During initial deployment, traffic profiles built upon SLAs between the service provider and a user are likely to be static on a PHB basis and change relatively infrequently. However as deployment expands, the model allows more dynamic negotiation on resource allocation between domains. Dynamic resource management can not alone be achieved without input of information from other components in a heterogeneous environment where each domain is administrated separately on different sets of policies. Hence our resource management frame work is based on the following components and we believe that such a structure can achieve our objectives more intelligently:

1. Traffic Measurement
2. Signaling
3. Route management

Traffic Engineering has a greater role in achieving dynamic resource management within the proposed architecture. Within this we stress implementing a measurement based resource allocation where the BB (PDP) monitors its current level of resource utilization and makes requests for additional bandwidth when resource utilization reaches some thresholds. Our architecture achieves this by implementing RIO [17] for the queue management within the router in the forwarding path. Edge routers monitor the traffic flows and tag the packets as being *in* or *out* of their profiles indicating drop priorities. In a measurement based resource allocation scheme, we stress performing traffic measurement at different time scales in order to avoid more processing time of the network devices. The time scale can be selected on the basis of which model we are implementing. For a provisioning model, timescale for traffic measurement can be in weeks while outsourcing model should consider in terms of hours only. In a direct model the time scale may be taken for few seconds only.

Signaling is equally important to achieve a better resource management within the Diff-serv networks. In our implementation, we propose to use RSVP between the user network and the ISP while we propose using COPS between the PDP and PEP within the ISP. Within a SAC we propose COPS signaling should be used, and for inter-domain SAC signaling then BGRP (Border Gateway Reservation Protocol) and/or SIBBS be used [18].

Route management is considered an important function within the resource management framework to achieve optimum resource utilization within the Diff-serv network and more specifically for the Inter-domain case. This is possible with proper route selection for traffic flows with different PHBs and can be achieved through load balancing amongst all the links between the Ingress and Egress pairs.

Apart from the above three functions, from business perspectives then Billing and Pricing between different service providers also have major roles during the process of resource negotiation and in future we propose to implement them within our frame work.

4 CONCLUSION AND FUTURE WORK

We have proposed a three-tier policy based resource allocation architecture to support end-to-end QoS, with a top level tier of an Inter Network Policy Database (PDB), a middle tier of a Service Provider PDB, and a lower tier of a Customer PDB. The alternative to policy based architecture

is constraint based resource allocation, but we propose that the decision on whether resources are available should be pushed to the edge router's CAC process [8]. This is consistent with the philosophy of Diff-serv which is to push the complexity of traffic classification, policing and shaping from the core of the network towards the edge.

We have considered administrative control, traffic engineering and resource management in particular, and within this architecture we have identified a User Administrative Control (UAC) to manage resources at the host, and two components of a Service provider Administrative Control (SAC): inter-domain and intra-domain. Each domain has a Bandwidth Broker (BB), which is the Policy Decision Point (PDP) and SAC, and is bounded by edge routers, which are the Policy Enforcement Points (PEPs) [5].

Future work is to implement a BB on our test bed, initially to support the Provisioning model, then to add the Direct-request and Out-sourcing model to support dynamic resource allocation.

4 REFERENCES

[1] A. Terzis, J. Ogawa, S. Tsui, L. Wang, L.Zhang; A prototype implementation of the two-tier architecture for differentiated services, RTAS 99, Vancouver, Canada, 1999

[2] A.J.Simmonds, P.Nanda; Resource management in differentiated services networks, Proceedings of 6th International symposium on communications Interworking,IFIP, Perth, 2002

[3] D. Goderis et al; A service-Centric IP QoS architecture for Next generation Networks, Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS'02), Florence, Italy, April 2002

[4] E. Mykoniati, D. Goderis, D. Griffin, P. Georgatsos; Admission Control for Providing QoS in Diffserv IP Networks: The TEQUILA approach, IEEE Communications, vol 41 no 1, Jan 2003

[5] F. Flegkas, P. Trimintzios, G. Pavlou. I Andrikopoulos, C.F. Cavalcanti; On Policy-based extensible Heirarchical Network management in QoS-enabled IP Networks, Proceedings of the IEEE Workshop on Policies for Distributed Systems and Networks (Policy '01), Bristol, UK, pp. 230-246, Springer, January 2001

[6] F. Wang, P. Mohapatra, S. Mukherjee, D. Bushmitch; An efficient bandwidth management scheme for real-time Internet applications, Computer Communications vol.25, 2002

[7] K. Nicholson, V. Jacobson, L.Zhang; A two-bit differentiated services architecture for the Internet, RFC 2639, 1999

[8] Ming Li, D.B. Hoang, A.J. Simmonds; Class based Fair Intelligent Admission Control over an Enhanced Differentiated Service Network, Proceedings of the International Conference on Information Networking 2003, Vol II, pp 783 - 792, ICOIN 2003, Jeju Island, S. Korea

[9] P. Trimintzios, P. Flegkas, G. Pavlou, L. Georgiadis, D. Griffin; Policy based Network dimensioning for IP differentiated services networks, Proceedings of the IEEE Workshop on IP Operations and Management (IPOM 2002), Dallas, Texas, USA, pp. 171-176, IEEE, October 2002.

[10] P. Nanda, A.J. Simmonds and S. Lee; Measuring Quality of Services in a Differentiated services domain with Linux, Proceedings of 5th International conference on Information technology, CIT-2002, Bhubaneswar, India, Tata Mcgraw-Hill publication, India

[11] P. Nanda, A.J. Simmonds; Providing End-to-End guaranteed Quality of Service over the Internet: A survey on Bandwidth Broker Architecture for Differentiated Services Network, Proceedings of 4th International conference on Information Technology,CIT-2001, Gopalpur, India, December 2001, Tata Mcgraw-Hill publication, India

[12] R. Rajan, D. Verma, S. Kamat, E. Felstaine, S. Herzog; A policy framework for Integrated and Differentiated Services in the Internet, Special Issue of IEEE Network Magazine on Integrated and Differentiated Services in the Internet, September 1999.

[13] R. Yavatkar, D. Pendarakis, R. Guerin; A framework for policy-based admission control, RFC 2753, January 2000

[14] R Rajan, D Durham, J Boyle, R Cohen, S Herzog, A Sastry; The COPS (Common Open Service Policy) Protocol, RFC 2748, January 2000

[15] S. Salsano; COPS usage for Diffserv Resource allocation(COPS-DRA), Internet Draft, October 2001

[16] Z.L. Zhang, Z. Duan and Y. Thomas; on scalable design of Bandwidth Brokers; IEICE Transaction in communication, vol.E84-B, No. 8, August 2001, pp 2011-2025.

[17] D.Clark, W.Fang, Explicit allocation of best effort packet delivery service, IEEE/ACM Transaction on Networking 6, no.4, 1998

[18] Phil Chemento, Ben Teitelbaum,Simple Inter domain Bandwidth Broker Specification (SIBBS), Proceedings of the First Joint Internet2 /DOE,QoSWorkshop *QBone,* Texas, February 2000