Journal of Computer Science and Cybernetics, V.24, N.2 (2008), 152–160

IMPLICIT SIGNALING PROCEDURE AND DECISIVE CRITERION OF STATELESS ROUTERS IN DIFFSERV DOMAIN

NGUYEN HONG SON[1], LE HUU LAP[2]

[1]*Post and Telecom Institute Technology, Ho Chi Minh city, Vietnam*
[2]*Post and Telecom Institute Technology, Hanoi, Vietnam*

Abstract. Differentiated Services (DiffServ) scheme fails to give the actual way for end user applications to request specific QoS as there is no connection admission control. Core routers are requested to be in stateless mode in IP DiffServ Networks. It means that they are not ready to control clear paths between endpoints. This puts an obstacle on attempt to manage connections in the networks. Lack of connection management, it is difficult to ensure QoS to user applications. In this paper, we propose a method for managing connections in IP DiffServ by using implicit signaling and decisive criterion at routers. This approach provides a mechanism of effective resource reservation to improve performance in provisioning QoS in IP networks.

Tóm tắt. Ngày nay, DiffServ (Differentiated Service) là giải pháp chiếm ưu thế trong việc thực hiện QoS trong mạng IP. Tuy nhiên, kiến trúc này lại không chuẩn hóa cơ chế điều khiển chấp nhận kết nối, nên các ứng dụng phía đầu cuối không có cách nào để yêu cầu QoS theo mong muốn. Điều nổi trội trong mạng IP DiffServ là các router lõi (core router) hoạt động theo chế độ phi trạng thái, có nghĩa là không lưu giữ thông tin của từng luồng lưu lượng. Thay vì vậy, các router lõi chỉ chuyển lưu lượng đi theo từng lớp (class), mỗi lớp được xử lý theo từng PHB (Per Hop Behaviour) đã được xác định. Cách làm này sẽ gây khó khăn cho các nỗ lực quản lý kết nối giữa hai đầu cuối (end-to-end). Do đó, ảnh hưởng đến sự đảm bảo QoS trong mạng. Trong bài báo này, chúng tôi đề nghị một phương pháp để quản lý các kết nối trong DiffServ domain bằng cách dùng giải pháp báo hiệu không tường minh (implicit signaling) và các tiêu chuẩn quyết định (decisive criterions) tại các router. Báo hiệu không tường minh cho kết quả là sự cho phép hay không một kết nối được thiết lập qua DiffServ domain, trong đó không tồn tại một đường nối tường minh. Một kết nối được phép khi tất cả các router cần thiết bắc cầu cho kết nối đó chấp nhận dựa theo tiêu chuẩn quyết đinh cục bộ đã được đề xuất trong bài này. Giải pháp ở đây mở ra triển vọng cung cấp một cơ chế đăng ký tài nguyên và quản lý kết nối hiệu quả nhằm khắc phục các khiếm khuyết nói trên của kiến trúc DiffServ.

1. INTRODUCTION

Currently, Differentiated Services (DiffServ) architecture is considered to be the most prevalent solution of QoS provision in IP networks. Traffic in the DiffServ networks is classified at the network border by edge routers (ERs), which mark packets by assigning a value to the DiffServ Code Point field (DSCP, which is a 6-bit field in the IP header) [1], and perform policing and shaping operations. Having assigned to the ERs a number of tasks, the DiffServ model leaves the complexity at the network edges, requiring core routers (CRs) to perform only aggregate classification (based on the DSCP) and to apply the consequent PHB (Per Hop Behavior). With such an approach, CRs do not have to implement complicated tasks such as active packets management (e.g., parsing and dropping or remarking operations), and their duty is to forward packets at the highest possible speed [2].

However, the DiffServ architecture has not been standardized for the solution of dynamic resource reservation and traffic engineering. Recently, the RMD (Resources Management in DiffServ) framework defines a dynamic resource reservation scheme that can be used for the dynamic SLS (Service Level Specifications) provisioning in an edge-to-edge DiffServ domain [3, 4]. RMD still encounters some problems, such as multi-unit reservations problem in the case of applications with diverse resource needs, which calls for handing connections with more than a single reservation unit. One separate reservation request packet has to send for each unit of the multi-unit reservations. This can introduce loss in resource utilization. Other problem of RMD is the reservation transient problem that appears when transforming from resource probing to reservation update as well as when terminating reservations. This problem may cause overbooked situation in network or packet loss. Another method is described in [5, 6], called GRI (Gauge & Gate Reservation with Independent Probing), which performs admission control by using probing method and measuring current load. GRIP also meets with the reservation transient problem. In order to solve this problem, [7] introduced a stack whose goal is to compensate the lack of packets emitted in transient time. Stack can prevent overbook situation but costs loss in utilization. Another approach for admision control in DiffServ domain proposed in [8]. By centralizing the resource management and provisioning, core routers are relaxed of maintaining flow related information. This centralizing method uses the concept of bandwidth broker, which depicts a fiction where a central device manages a DiffServ domain. However, using bandwidth brokers introduces a sensitive single point of failure and a bottleneck in networks. In addition, some investigations on the possibility of incorporating DiffServ into MPLS networks are given in [9, 10]. This solution is regarded as a promising solution to perform traffic engineering in DiffServ networks. However, many problems still need to be solved by the incorporating approach.

As to DiffServ environment, we found that using measurement-based admission control (MBAC) may cause overload unless adding to it a special solution. Because DiffServ networks operate in connectionless mode, thus routers can make mistakes in acceptance new flows if it is only base on the current measured load for controlling connected admission. This can result in allowing excessive number of flows that cause overload in DiffServ domain. None of existing proposals paid attention to this problem in DiffServ environment. This is the reason why [3, 4, 5] encounter the above-mentioned drawbacks.

In this paper, we propose an approach for implicit signaling in cooperating with a decisive criterion at routers for controlling whether to admit a new flow. The approach will avoid the drawbacks of previous proposals by overcoming the above-mentioned problem. In this solution, we also try to preserve the scalability in DiffServ environment while manage resources effectively and protects current flows from QoS degradation. The designed signaling procedure is used by edge routers to request an allowance of communication between two hosts thru DiffServ domain. We also introduce a mathematical model for building a decisive criterion for DiffServ routers. Routers will base locally on the decisive criterion for processing requests. Operation of routers also includes measurement process [11], which estimates the load of existing traffic in router. This is because load of existing traffic is one of terms in the decisive criterion. Our method is developed to operate on each PHB separately in order to be compliant with DiffServ environment.

The organization of the paper is as follows. In Section 2 we design the signaling procedure that is used to register a connection thru DiffServ domain, our procedure is different from the probing procedure of [3, 4, 5]. We develop decisive criterion for routers that is suitable for the designed signaling procedure in Section 3. Section 4 ends the paper with conclusions.

2. THE IMPLICIT SIGNALING PROCEDURE IN DIFFSERV DOMAIN

Regarding hosts use a certain flow to communicate thru a DiffServ domain. Traffic of flows will be conditioned and marked and grouped into BAs (Behavior Aggregates) according to their Tspec (Traffic Specifications). Two hosts need to get the agreement of DiffServ domain about putting a flow between them across the domain before transmitting data. In this section, we propose a signaling procedure that is used by edges to set up an implicit connection in DiffServ domain. Essentially, the signaling procedure is between ingress and egress that aims to ask DiffServ domain whether to have enough resources for a new flow. While RSVP is well done using in IntServ architecture, it is not suitable for DiffServ environment as there is requirement of information storage at core router for explicit paths. In our proposed procedure, ingress doesn't ask for an explicit path. It means that the ingress only cares the right of putting a new flow in DiffServ domain and no need to know how to carry data traffic of the new flow inside domain.

Note that the number of accepted flows in DiffServ domain can increase excessively if decisive criterion of routers bases only on current measured load in the connectionless operation mode. Because in this operation mode, the core routers accept a new flow but actual load of that flow may be routed to another, and the router can continue to allow more flows. This will result overload in DiffServ domain. Thus, it is necessary to add one more rule into decisive criterion for limiting the excessive increase of number of flows entering DiffServ domain. The added rule is based on the number of current allowed flows in routers. For creating favorable conditions, DiffServ routers have to keep track of the number of flows accepted by them, say n. It means that the routers only hold and update their parameter n, or the procedure does not ask DiffServ routers to keep per flow information.

Signaling procedure operates between ingress router and egress router in order to probe DiffServ domain whether to allow a new flow. Ingress does not ask for an explicit path, what path for new flow is managed inside domain. Figure 1 illustrates operation of the proposed signaling procedure in the case user's request is accepted.

The procedure uses some control packets such as:

- Request packet: on receiving a user request, ingress router issues a request packet, which contains type of packet, addresses, request bandwidth, a flag bit, and counting number. Role of the counting number is to keep trace of number of routers that accepted the request.
- Accept packet: egress router informs ingress router about acceptance, the packet contains addresses, type of packet.

- Reject packet: egress router informs ingress router about no acceptance, the packet contains addresses, type of packet, and counting number.
- Release packet: ingress router issues this packet for releasing resources when it finishes transmitting data. This makes each relevant router to decrease its parameter n by one.
- Clear packet: on receiving a reject packet, ingress router copies the counting number to TTL of the clear packet and sends it to domain for releasing resources at routers which accepted the request packet before the packet was marked.

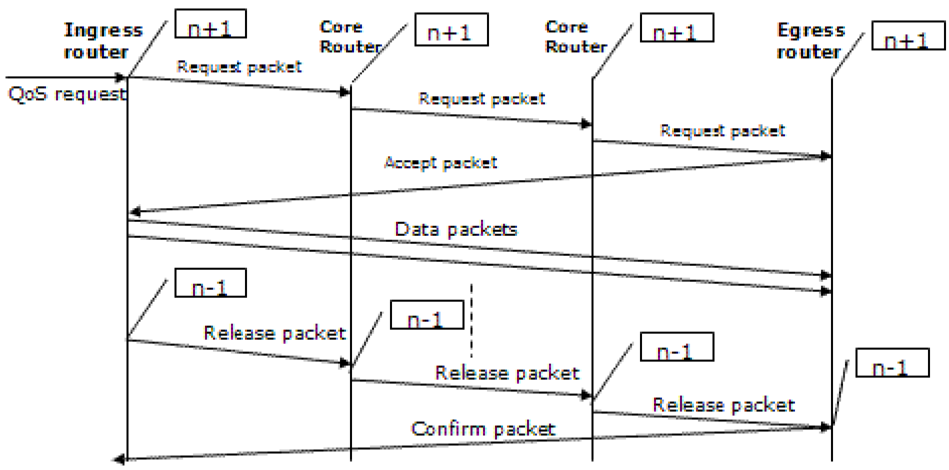

Figure 1. User's request is accepted

Note that the proposed method is performed on each PHB separately. Thus, control packets also contain DSCP code to indicate PHB corresponding to the request of end user application. Followings are operating regulations of the signaling procedure:
- User sends request packet to ingress router when need to connect with certain destination.
- On receiving a request packet, ingress router checks the request with the decisive criterion; if satisfying the rule, it takes three jobs: increasing counting number of the packet by one, increasing its parameter n by one, and forwarding the packet to next router. On the contrary, it responds to user application with a reject packet, and cancels the packet.
- On receiving a non-marked request packet, core router checks the request with decisive criterion; if satisfying the rule, it takes three jobs: increasing counting number of the packet by one, increasing its parameter n by one, and forwarding the packet to next router. On the contrary, it marks a request packet by setting flag bit, and then forwarding the packet to next router.
- On receiving a marked request packet, core routers just forward the packet to next router.
- On receiving a marked request packet, egress router just responds to ingress router with a reject packet; it also copies counting number from marked request packet to the reject packet.
- On receiving a non-marked request packet, egress router also checks the request with decisive criterion. If satisfying the rule, the router takes two jobs: increasing its parameter n by one and responding to ingress router with an accept packet. On the contrary, it responds to ingress router with a reject packet; it also copies counting number from marked request packet to the reject packet.

- On receiving an accept packet from egress router, ingress router informs end user application for starting to transmit data to domain.
- Ingress router will issue a release packet to domain when finishes transmitting data.
- On receiving a reject packet, ingress router makes a clear packet, copies counting number from reject packet to TTL of the clear packet, and sends the clear packet to domain for releasing resources at routers which accepted the request packet before the packet was marked.
- On receiving clear packet or release packet, router decreases its parameter n by one.

As an example, Figure 2 presents operation of signaling procedure in an unsuccessful case.

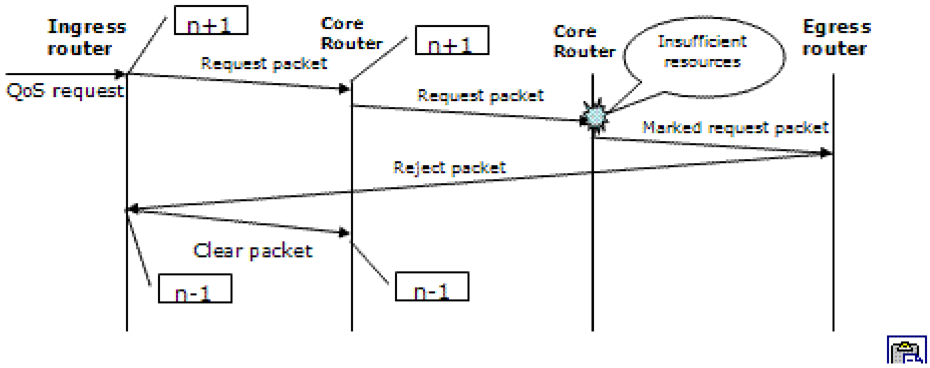

Figure 2. User's request is rejected

3. THE DECISIVE CRITERION OF STATELESS ROUTERS

Let X be a sum of n independent random variables $\{X_i\}$, X_i is non-negative. Following Chernoff bound a distribution's tail, we have

$P\{X \geq \mu\} \leq E[e^{s(X-\mu)}]$,

where $s > 0$,

$\ln P\{X \geq \mu\} \leq \ln E[e^{s(X-\mu)}]$,

$\ln P\{X \geq \mu\} \leq s. \ln E[e^X e^{-\mu}]$,

$\ln P\{X \geq \mu\} \leq s(\ln E[e^X] - \mu)$,

$\ln P\{X \geq \mu\} \leq s(\frac{1}{s} \ln E[e^{sX}] - \mu)$,

Let $y(s) = \dfrac{1}{s} \ln E[e^{sX_i}]$.

All flows entering DiffServ router originate from ON-OFF sources. Here, we consider the case of $X_i = p_i$ during ON and $X_i = 0$ during OFF, with p_i and m_i are peak rate and average rate of a flow i $(i = 1, 2, 3...n)$, respectively. We also regard probability of X_i, p_i, as source activity probability or $P\{X_i = p_i\} = m_i/p_i$, and $P\{X_i = 0\} = 1 - m_i/p_i$. Therefore,

$$y(s) = \frac{1}{s} \ln[(1 - \frac{m_i}{p_i})e^0 + \frac{m_i}{p_i} e^{sp_i}],$$

$$y(s) = \frac{1}{s} \ln[1 + \frac{m_i}{p_i}(e^{sp_i} - 1].$$

We claim that for every real x, $e^x \geq 1 + x$, so that

$$y(s) = \frac{1}{s}\ln[1 + \frac{m_i}{p_i}(e^{sp_i} - 1] \leq \frac{1}{s}\ln e^{\frac{e^{sp_i-1}}{p_i}m_i}.$$

Then

$$y(s) \leq \frac{e^{sp_i} - 1}{sp_i}m_i.$$

Hence, $\ln P\{X \geq \mu\} \leq -\varepsilon$ holds if there exists a number s such that

$$s.(\frac{1}{s}\ln E[e^{sX}] - \mu) \leq -\varepsilon.$$

Parameter ε takes the role of requested level of QoS.

$$\frac{1}{s}\ln E[e^{sX}] + \frac{\varepsilon}{s} \leq \mu,$$

$$\sum_{i=1}^{n} \frac{e^{sp_i-1}}{sp_i}m_i + \frac{\varepsilon}{s} \leq \mu. \tag{$*$}$$

The left-hand side of inequality (*) has minimum at some value s with

$$\varepsilon = \sum_{i=1}^{n} \frac{m_i}{p_i}.(sp_i e^{sp_i} - e^{sp_i} + 1). \tag{$**$}$$

Substituting (**) into (*), we get

$$\sum_{i=1}^{n} e^{sp_i}m_i \leq \mu, \text{ for all } s > 0. \tag{$***$}$$

In the case of DiffServ domain, each router will treat flows depending on PHB of flows. Each PHB is assigned fixed number of resources, call μ, so here we can focus on decisive criterion of connection admission for every PHB only. As we know, all flows in a BA (behavior aggregate) have the same peak rate determined by the same token bucket in their DiffServ class, so $p_1 = p_2 = ... = p_n = p$. Let $\hat{\mathsf{S}}$ be the measured load of existing traffic inside a certain PHB of a router, and let r be the request load of a new flow, apply (***) to this case we have

$$e^{sp}(\hat{\mathsf{S}} + r) \leq \mu, \ s \geq 0.$$

Let $f = \mu/e^{sp}$, f reaches maximum at $s = 0$. Hence, s takes the role of control parameter to govern the possible maximum value of $(\hat{\mathsf{S}} + r)$. In other words, s stipulates the limit of the bandwidth to be used, or $(\hat{\mathsf{S}} + r)_{\max} = v.\mu$.

$$v = \frac{1}{e^{sp}}, \quad s = \frac{\ln v^{-1}}{p}.$$

Here, v is a coefficient of using the bandwidth μ, $(0 < v \leq 1)$.

Since the parameter $\hat{\mathsf{S}}$ is estimated by (1), it may be not equal to the reserved traffic load at a node. Indeed, let $u_i(t)$ be the reserved rate of flow i of a class at a time t, the S_r reserved in a sample time T is

$$S_r = \frac{X_T}{T}$$

or

$$S_r = \frac{1}{T}.\int_0^T \sum_{i=1}^{n} u_i(t)dt.$$

Let a_i be the probability of presence of flow i in T, we have

$$\hat{\mathsf{S}}_E = \frac{\sum_{i=1}^{n} a_i u_i T}{T} = \sum_{i=1}^{n} a_i u_i.$$

Obviously, $\hat{\mathsf{S}}_E \leq S_r$.

If $\hat{\mathsf{S}}_E$ is much smaller than S_r, a router may make decisions, which causes it to be overload in near future when all allowed flows transmit data simultaneously. Moreover, as mentioned above, the number of accepted flows will increase excessively in domain if the decisive criterion of connection admission bases only on current measured load in connectionless mode. This can result in overload in DiffServ domain. Thus, it is necessary to protect domain from both overload situations by putting a more constraint on the number of accepted flows as follows

$$n \leq \frac{\mu}{\alpha.r_m},\ 0 < \alpha \leq 1$$

where r_m is the mean rate of traffic source of user whom want to use this service class.

Let $\delta = (1 - v)\mu$, be the allowed deviation of μ. We can determine the value α as:

$$\frac{\mu + \delta}{r_m} = \frac{\mu}{\alpha.r_m}$$

or

$$\alpha = \frac{1}{2 - v},\ 0 < v \leq 1.$$

Finally, the decisive criterion at DiffServ router for each class is defined by

$$(n+1) \leq \frac{\mu}{\frac{1}{2-v} r_m},\ e^{sp}.(\hat{S} + r_m) \leq \mu,\ s > o,\ 0 < v < 1$$

or

$$(n+1) \leq \frac{\mu}{\frac{1}{2-e^{sp}} r_m},\ e^{sp}.(\hat{S} + r_m) \leq \mu,\ s > o,$$

where

n: number of accepted flows in router,

μ: output capacity assigned to specific DiffServ class at router,
$\hat{S}$: current estimated load,
r_m: mean rate of traffic source,
p: peak rate,
s: control parameter.

Note that, as mentioned in the previous section, each router holds a parameter n to indicate current number of accepted flows, a new flow with r_m is accepted if n and r_m satisfy simultaneously the above inequality system. Otherwise, a new flow is rejected. The parameter s takes the roles of control parameters in this decisive criterion and determined according to v.

4. CONCLUSION

In order to overcome lacks of DiffServ mechanism, we have proposed a method for performing admission control in DiffServ environment. These techniques are implicit signaling and decisive criterion, which were presented in this paper. It gives end user applications a way to reserve resources and prevent IP networks from degradation of QoS provision. Unlike the previous methods, the signaling procedures do not set up an explicit connection because core routers are stateless. The locally decisive criterion of routers based on not only measured load of existing traffic in router, but also on the current number of accepted flows in the router. While providing a good mechanism of connection reservation management, our approach can avoid the drawbacks of others, such as GRIP, RMD. However, the performance of the method needs to be appraised more detail, especially in utilization and packet loss. This is our future work, and it will be presented in next papers.

REFERENCES

[1] K. Nichols, S. Blake, F. Baker, D. Black, Definition of the differentiated services field (DS field) in the IPv4 and IPv6 header, *IETF RFC 2474* (December 1998).

[2] B. Carpenter, K. Nichols, Differentiated services in the internet, *Proceedings of the IEEE* **90** (9) (2002) 1479–1494.

[3] L. Westberg, M. Jacobsson, G. Karagiannis, S. Oosthoek, D. Partain, V. Rexhepi, R. Szabo, P. Wallentin, Resource management in diffserv framework, internet draft, work in progress, 2001.

[4] Westberg, L., Karagiannis, G., Partain, D., Oosthoek, S., Jacobsson, M., Rexhepi, V., Resource Management in Diffserv On DemAnd (RODA) PHR, Internet Draft, Work in progress.

[5] G. Bianchi, N. Blefari-Melazzi, M. Femminella, F. Pugini, GRIP: Technical report, work in progress, http://drake.diei.unipg.it/netweb/GRIP_tech_rep.pdf).

[6] G. Bianchi, N. Blefari-Melazzi, "A migration path to provide end-to-end QoS over stateless networks by means of a probing-driven admission control", Work group: Internet Draft, November 2001.

[7] G. Bianchi, N. Blefari-Melazzi, M. Femminella, F. Pugini, *Performance evaluation of a measurement-based algorithm for distributed admission control in a diffServ framework, IEEE Globecom 2001*, San Antonio, Texas, USA, 25-29 November 2001.

[8] Jani Lakkakorpi, Ove Strandberg, and Jukka Salonen, Adaptive connection admission control for differentiated services access networks, *IEEE Journal On Selected Areas In Communications* **23** (10) (October 2005).

[9] F. Le Fraucheur, MPLS Support of Differentiated Services, *RFC3270*, May 2002.

[10] H. Jonathan Chao, Xiaolei Guo, *Quality of service control in high-speed networks*, John Wiley & Son, Inc, 2003.

[11] S. Jamin, P. B. Danzig, S. J. Shenker, and L. Zhang, A measurement-based admission control algorithm for integrated services packet networks, *Proc. of ACM SIGCOMM '95*, Cambridge, Massachusetts, USA, 1995 (2–13).

[12] A. Marquetant, O. Pop, R. Szabo, G. Dinnyes, Z. Turanyi, Novel enhancements to load control a soft state, lightweight admission control protocol, *QofIS'2000-2nd International Workshop on Quality of future Internet Services*, Berlin, Germany, September 2001.

[13] V. Firoiu, J. Y. Le Boudec, D. Towsley, Z. L. Zhang, Theories and models for internet quality of service, *Proceedings of the IEEE* **90** (9) (2002) 1565–1591.

[14] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, An architecture for differentiated services, *IETF RFC 2475*, December 1998.

Received on May 20 - 2008