Internet Engineering Task Force
INTERNET-DRAFT
Expires: March 1999

Borje Ohlman
Ericsson
Petri Koskelainen
Nokia

30 September 1998

Receiver control in Differentiated services

<draft-ohlman-receiver-ctrl-diff-01.txt>

Abstract

This draft addresses the issue of receiver control for the specific case where the receiver needs to control incoming traffic on its own access link. This is of particular importance for low bandwidth links.

1. Introduction

In Differentiated Services the TOS bits are set at the sender side of the network. At receivers access link this setting might not reflect how the receiver wants the incoming traffic prioritized. This draft discusses how this problem could be solved by applying a different semantics for the TOS bits at the last hop router compared to what is applied through the rest of the network. It is important to recognize that the receiver (the owner) of access link must always be capable of fully controlling the the usage of that access link.

2. Types of receiver control

The concept of receiver control can be applied to (at least) two different contexts. One is when the receiver is allowed to control which priority should be set by the sender (this can be of interest for a user who is eager to get the result from a http request delivered promptly). Another type is when the receiver need to control the priority of the packets that comes from the network onto his/her access link. Two reasons exists why this is important. One is that it provides protection from certain types of denial of service attacks. The other is that this is important on low bandwidth access links, in particular for cellular IP hosts.

This proposal does not try to address the first type of receiver control. We simply note that this problem can initially be solved at the session or application layers. It still remains to be shown that a mechanism is needed at the network layer.

This proposal suggests a method for giving the receiver control over his/her access link. Note that there exists two cases of this access link issue. First there is last-hop access link issue, and secondly there is organization's access link issue.

3. Motivation for access link receiver control

The problem with low bandwidth access links are not going away within the foreseeable future. In addition to an expected continued use of dial-up modem connections over POTS we expect to see large number of mobile phones becoming IP hosts.

Example 1: Receiver control in an organization's access link.

This problem arises e.g. between a company LAN and an ISP.
For example, a company may have a 64 Kbit/s incoming link
from an ISP. If a company worker is surfing the web and clicks
on an advertisement video button (e.g. about new car model)
which consumes ~100 Kbit/s (marked with highest possible DS value).
One such advertisement video will utilize whole access link
capacity thus preventing all other activities in that link.
E.g. if the company CEO wants to download something urgently or to
use an IP phone application he/she is unable to do so.

Example 2: Receiver control in low bitrate last-hop link.

A user is having an IP-phone conversation with a person. This person then asks our user to look at a web page. The web page our user requests comes from a commercial web server which prides itself of always giving prompt responses, this includes sending all traffic as

high priority.
If the IP-phone conversation is only using best effort it might be severely degrade by the http-download, this is most likely not what our user would like.

In the current diff-serv discussions the focus seems to be on having edge devices rather than users/end-systems setting the TOS-bits. Assuming this setting is done according to service level agreements with the senders ISPs, the receiver has no way of influencing the way the traffic is prioritized.

The low priority of the IP-phone traffic is not a problem through a lightly loaded backbone network. It is not until it is merged with the web download on the low-bandwidth access link to the receivers laptop a significant delay occurs. If the receiver could control how the traffic is prioritized over the narrow access link this could easily be solved.

These two similar kind of problems exists at
the edge of differentiated services Internet.

They can be solved by receiver control, depending
on the needed flexibility it can be achieved in
two ways:

1) Static configuration (receivers have beforehand
defined some policies, e.g. packets from company.com
are always the most important packets)

2) Dynamic configuration (signalling)

In many situations static configuration will be too
limited. Then some kind of signalling is needed.
Although DiffServ is based on "no signalling" approach,
this signaling should not affect the diff serv world
since it is applied purely in the end user network
(access link or end users low bitrate last-hop link).
Basically it is no different than using RSVP at the edges.

4. Suggested solution

To give the receiver control over which flows it values most we suggest that the semantics of the priority bits is changed across the receivers access link, compared to within the network. Instead of defining the priority over the access link they will be regarded as a request. This will mean that the access node will not grant priority according to TOS bits unless they are in agreement with the receivers wishes. The receivers preferences can either be expressed via static

configuration in a user profile or the user can be prompted for each new incoming flow.

This gives the user the power to control the incoming traffic on his/her access link.

The idea is that DiffServ marked flows are treated like best-effort flows in access link unless otherwise ordered by the receiver.

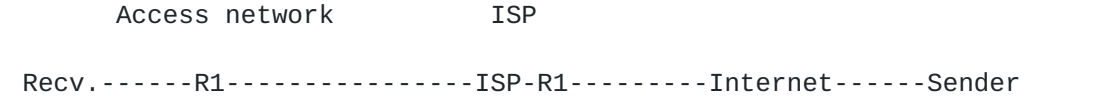
The receiver acknowledges the request (DS-marked best-effort packet itself is a requests) to next-hop router whenever it wants to accept the DiffServ flow.

Otherwise the flow continues as a Best-Effort flow. The receiver may send the Ack beforehand (e.g. when starting the application level signalling like SIP, H.323, WWW clicking etc) or when it gets the first DS-marked packet (which cames as a BE packet).

ACK messages should be forwarded upstream until they reach a router that has not been configured as a "last hop router", i.e. it does not understand the ACK message.

A router may be configured to forward ACK messages upstream (e.g. last-hop router may forward ACK to corporate access router).
Receiver must send the ACK always to its closest router. These forwarding issues can be defined in service level agreements (SLAs).
Router can be defined to accept and/or forward ACK messages.
Router which is not allowed to accept ACK messages must silently discard those.

Example:



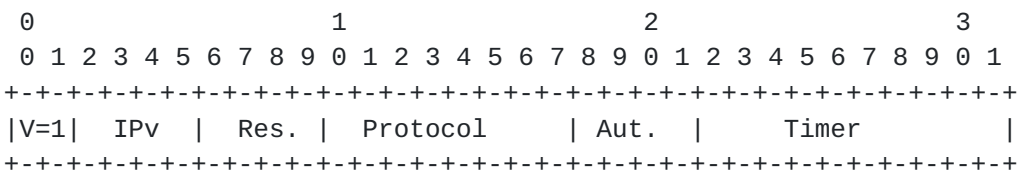
1. Sender starts sending packets with highest possible DiffServ values
2. Packets travel through DS-capable network
3. Packets reach ISP-R1 which forwards packets as best-effort to access link.
4. Receiver gets first packets and it sends ACK to R1. Now R1 honours DS-values for that flow.
5. R1 sends ACK to ISP-R1 (which is propably the actual bottleneck). Now ISP-R1 accepts incoming DS-packets for that flow (instead of treating those as BE packets).
6. Packets from sender to destination are handled everywhere as they should be handled. Receiver makes periodically ACK refresments.

Following chapter defines the packet format for this simple dynamic DiffServ Ack Protocol (DAP).

5. Packet Format

This ACK packet is sent to special UDP port (TBD). There is not any negotiation involved but the refreshment timer can be defined by the receiver. The packet format is as follows:

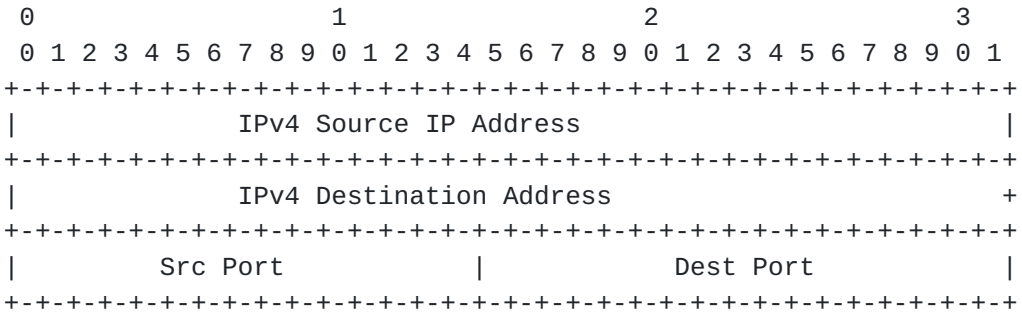
Fixed format:



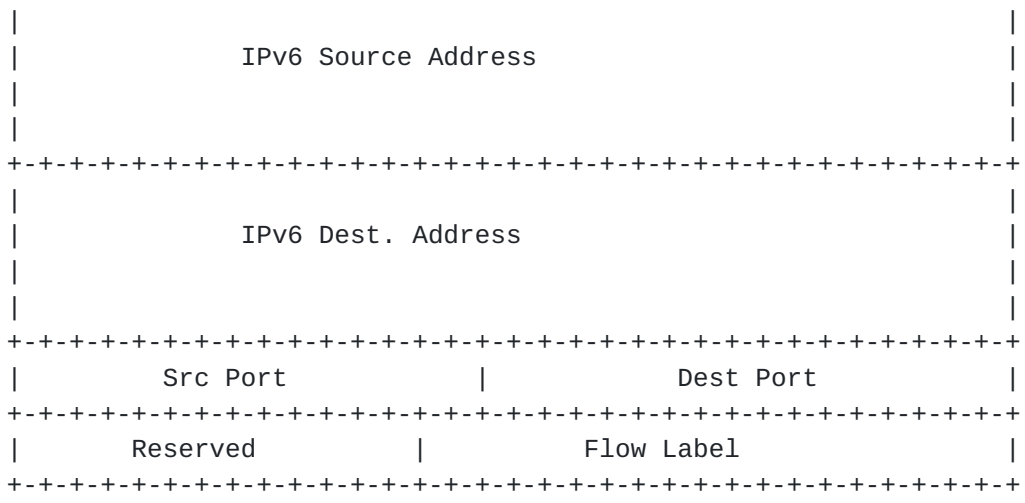
V: 2 bits. Version number. Current version is 1.
IPv: 4 bits. IP version number. Currently IPv4 and IPv6 are defined.
Reserved: 4 bits . Reserved for future use.
Protocol: 8 bits. The protocol number (usually UDP).
Aut: 4 bits. Authentication is in use. Currently only Aut=1 is defined (means RSVP User Identity).
Timer: 10 bits. This contain the value in seconds for the ACK refresh period. If router does not get any ACK for some flow in (timer*2) seconds then DS-enabling state is released. Maximum refresment interval is 1024 seconds (about 17 minutes). If timer=0 then refresments are not used at all.

Depending on the IP version number, following additional header is included. Wildcarding is allowed (using zeros).

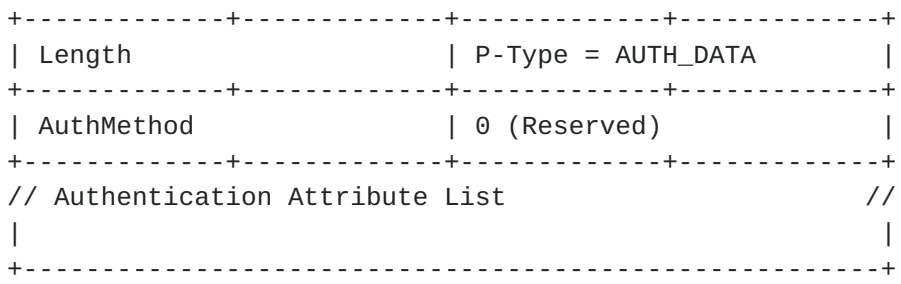
Case IPv4:



Case IPv6:



It is also possible to have authentication header included. In that case Aut=1 and the rules for RSVP user identity [4] are followed.



6. Denial of service attacks

If someone is trying to flood your access link with high priority packets (e.g. aggressive marketers), the above suggested mechanism can help the user to make sure those packets are dropped at the last hop router (or in corporate access link router), thus protecting the preferred traffic on a low bandwidth link.

7. Considerations

In a lot of cases it is expected that source address will provide enough information for meaningful flow identification (which allows this mechanism to work also with end-to-end encrypted traffic).

The amount of traffic dropped in an overload situation is the same with or without this mechanism, the difference is that this gives the receiver a better chance of influencing which traffic is dropped.

8. Conclusion

This draft points out that there are variants on the concept of

receiver control. This should be reflected in the diff-serv framework document. It should also be clarified which of these aspects the current diff-serv work intends to address.

To make it possible for the receiver to control its own access link it is important that the diff-serv standard allows for the last hop router to priorities traffic in accordance with the receivers requests even if this is in contradiction with the TOS-bits settings. To achieve this the TOS bits should only be regarded as requests at the last hop router.

References

[1] D. Clark and J. Wroclawski, "An Approach to Service Allocation in the Internet", Internet Draft <draft-clark-diff-svc-alloc-00.txt>, July 1997.

[2] K. Nichols, V. Jacobson, and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet", Internet Draft <draft-nichols-diff-svc-arch- 00.txt>, November 1997.

[3] K. Nichols and S Blake, "Differentiated Services Operational Model and Definitions", Internet Draft <draft-nichols-dsopdef-00.txt>, February 1998.

[4] S. Yadav, R. Pabbati, P. Ford and S. Herzog, "User Identity Representation for RSVP", Internet Draft <draft-ietf-rap-user-identity-00.txt>, March 1998.

Author's Address

Borje Ohlman
Ericsson
Varuvägen 9 (Älvsjö)
S-126 25 Stockholm
Sweden

Phone: +46-8-719 3187
Fax. +46-8-719 6677
E-mail: Borje.Ohlman@ericsson.com

Petri Koskelainen
Nokia Research Center
P.O. 100
FIN-33721 Tampere
Finland

E-mail: petri.koskelainen@research.nokia.com