V6OPS	B. E. Carpenter
Internet-Draft	Univ. of Auckland
Intended status: Informational	S. Jiang
Expires: April 15, 2012	Huawei Technologies Co., Ltd
	October 13, 2011

Using the IPv6 Flow Label for Server Load Balancing
draft-carpenter-v6ops-label-balance-00

Abstract

This document describes how the IPv6 flow label can be used in support of layer 3/4 load balancing for large server farms.

Status of this Memo

Copyright Notice

Table of Contents

*1. Introduction

*2. Role of the Flow Label

*3. Security Considerations

*4. IANA Considerations

- 5. Acknowledgements
- 6. Change log [RFC Editor: Please remove]
- 7. References
- 7.1. Normative References
- 7.2. Informative References
- Authors' Addresses

1. Introduction

The IPv6 flow label has been redefined [I-D.ietf-6man-flow-3697bis] and its use for load balancing in multipath routing has been specified [I-D.ietf-6man-flow-ecmp]. Another scenario in which the flow label could be used is in load balancing for large server farms. This document starts with a brief introduction to load balancing techniques and then describes how the flow label can be used to enhance layer 3/4 flow balancers in particular.

Load balancing for server farms is achieved by a variety of methods, often used in combination [Tarreau]. The flow label is not relevant to all of them. Also, the actual load balancing algorithm (the choice of server for a new client session) is irrelevant to this discussion.

- The simplest method is simply using the DNS to return different server addresses for a single name such as www.example.com to different users. Typically this is done by rotating the order in which different addresses are listed by the relevant authoritative DNS server, assuming that the client will pick the first one. The flow label can have no impact on this method and it is not discussed further.

- Another method, for HTTP servers, is to operate a layer 7 reverse proxy in front of the server farm. The reverse proxy will present a single IP address to the world, communicated to clients by a single AAAA record. For each new client session (an incoming TCP connection and HTTP request), it will pick a particular server and proxy the session to it. Hopefully the act of proxying will be cheap compared to the act of serving the required content. The proxy must retain TCP state and proxy state for the duration of the session. This TCP state could, potentially, include the incoming flow label value.

- A component of some load balancing systems is an SSL reverse proxy farm. The individual SSL proxies handle all cryptographic aspects and exchange raw HTTP with the actual servers. Thus, from the load balancing point of view, this really looks just like a server farm, except that it's specialised for HTTPS. Each proxy will retain SSL and TCP and maybe HTTP state for the duration of the session, and the TCP state could potentially include the flow label.

*Finally the "front end" of many load balancing systems is a layer 3/4 load balancer. In this case, it is the layer 3/4 load balancer whose IP address is published as the primary AAAA record for the service. All client sessions will pass through this device. According to the precise scenario, it will spread new sessions across the actual application servers, across an SSL proxy farm, or across a set of layer 7 proxies. In all cases, the layer 3/4 load balancer has to recognize incoming packets as belonging to new or existing client sessions, and choose the target server or proxy so as to ensure persistence. 'Persistence' is defined as guaranteeing that a given session will run to completion on a single server. The layer 3/4 load balancer, whatever method it uses for forwarding the session, is certain to inspect the source address and the protocol and port numbers in each incoming packet. At the same time, it could inspect and make use of the flow label.

Layer 3/4 load balancers use various techniques to actually reach their target server.

- All servers are configured with the same IP address, they are all on the same LAN, and the load balancer sends directly to their individual MAC addresses.
- Each server has its own IP address, and the balancer uses an IP-in-IP tunnel to reach it.
- Each server has its own IP address, and the balancer performs NAPT (address and port translation).

The following diagram, inspired by [Tarreau], shows a maximum layout.

```
     ___________________________________________
    (                                           )
    (          Clients in the Internet          )
    (___________________________________________)
                         |
                    ------------
                    | Ingress  |
                    | router   |
                    ------------
             ____________|_____________
             |                        |
             |DNS-based load splitting|
             |                        |
        ------------             ------------
        |L3/L4 ASIC|             |L3/L4 ASIC|
        | balancer |             | balancer |
        ------------             ------------
             |          load          |
             |        spreading       |
   __________|________________________|___________
       |              |            |          |
 ------------   ------------   --------   --------
 |HTTP proxy|...|HTTP proxy|   | SSL  |...| SSL  |
 | balancer |   | balancer |   | proxy|   | proxy|
 ------------   ------------   --------   --------
   ____|_____________|_____________|_________|_____
     |          |          |          |          |
 --------   --------   --------   --------   --------
 |HTTP  |   |HTTP  |   |HTTP  |   |HTTP  |   |HTTP  |
 |server|   |server|   |server|   |server|   |server|
 --------   --------   --------   --------   --------
```

From the previous paragraphs, we can identify several points in this diagram where the flow label may be relevant:

1. L3/L4 load balancers.

2. SSL proxies.

3. HTTP proxies.

2. Role of the Flow Label

The IPv6 flow label is included in every IPv6 header [RFC2460] and it is defined in [I-D.ietf-6man-flow-3697bis]. According to this definition, it should be set to a constant value for a given traffic flow (such as an HTTP connection), but until the standard is widely implemented it will often be set to the default value of zero. Any device that has access to the IPv6 header has access to the flow label, and it is at a fixed position in every IPv6 packet. In contrast, transport layer information, such as the port numbers, is not always in a fixed position, since it follows any IPv6 extension headers that may be present. Therefore, within the lifetime of a given transport layer connection, the flow label can be a more convenient "handle" than the port number for identifying that particular connection.

According to [I-D.ietf-6man-flow-3697bis], source hosts should set the flow label, but if they do not (i.e. its value is zero), forwarding nodes may do so instead. In both cases, the flow label value must be constant for a given transport session, normally identified by the IPv6 and Transport header 5-tuple. The flow label should be calculated by a stateless algorithm. The value should form part of a statistically uniform distribution, making it suitable as part of a hash function used for load distribution. Because of using a stateless algorithm to calculate the label, there is a very low (but non-zero) probability that two simultaneous flows from the same source to the same destination have the same flow label value despite having different transport protocol port numbers.

The suggested model for using the flow label in a load balancing mechanism is as follows.

- It is clearly better if the original source, e.g. an HTTP client, sets the flow label. However, if the flow label of an incoming packet is zero, the ingress router at the server site should implement the stateless mechanism in Section 3 of [I-D.ietf-6man-flow-3697bis] to set the flow label value to an appropriate value. This relieves the subsequent load balancers of the need to fully analyse the IPv6 and Transport header 5-tuple.

- The L3/L4 load balancers use the 2-tuple {source address, flow label} as the session key for whatever load distribution algorithm they support, instead of searching for the transport port number later in the header. This means they can ignore all IPv6 extension headers, which should simplify their design and lead to a performance benefit.

- The SSL proxies may do the same. However, since they have to process the transport layer in any case, this might not lead to any performance benefit.

- The HTTP proxies may do the same. However, since they have to process the transport and application layers in any case, this might not lead to any performance benefit.

Note that in the unlikely event of two simultaneous flows from the same source having the same flow label value, the two flows would end up assigned to the same server, where they would be distinguished as normal by their port numbers. Since this would be a statistically rare event, it would not damage the overall load balancing effect.

3. Security Considerations

Security aspects of the flow label are discussed in [I-D.ietf-6man-flow-3697bis]. As noted there, a malicious source or man-in-the-middle could disturb load balancing by manipulating flow labels. Specifically, [I-D.ietf-6man-flow-3697bis] states that "stateless classifiers should not use the flow label alone to control load distribution, and stateful classifiers should include explicit methods to detect and ignore suspect flow label values." The former point is answered by also using the source address. The latter point is more complex. If the risk is considered serious, the ingress router mentioned above should verify incoming flows with non-zero flow label values. If a flow from a given source address and port number does not have a constant flow label value, it is suspect and should be dropped.

4. IANA Considerations

This document requests no action by IANA.

5. Acknowledgements

Valuable comments and contributions were made by
This document was produced using the xml2rfc tool [RFC2629].

6. Change log [RFC Editor: Please remove]

draft-carpenter-v6ops-label-balance-00: original version, 2011-10-13.

7. References

7.1. Normative References

[RFC2460]	Deering, S.E. and R.M. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.
[I-D.ietf-6man-flow-3697bis]	Amante, S, Carpenter, B, Jiang, S and J Rajahalme, "IPv6 Flow Label Specification", Internet-Draft draft-ietf-6man-flow-3697bis-07, July 2011.

7.2. Informative References

[RFC2629]	Rose, M.T., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.
[I-D.ietf-6man-flow-ecmp]	Carpenter, B and S Amante, "Using the IPv6 flow label for equal cost multipath routing and link

	aggregation in tunnels", Internet-Draft draft-ietf-6man-flow-ecmp-05, July 2011.
[Tarreau]	Tarreau, W. , "Making applications scalable with load balancing", 2006.

Authors' Addresses

Brian Carpenter Carpenter Department of Computer Science University of Auckland PB 92019 Auckland, 1142 New Zealand EMail: brian.e.carpenter@gmail.com

Sheng Jiang Jiang Huawei Technologies Co., Ltd Q14, Huawei Campus No.156 Beiqing Road Hai-Dian District, Beijing, 100095 P.R. China EMail: jiangsheng@huawei.com