COIN
Internet-Draft
Intended status: Standards Track
Expires: June 11, 2020

H. Singh
MNK Consulting

December 9, 2019

# New IPv6 Multicast Addresses for Switch ML
# draft-hsingh-ipv6-coin-ml-03

Abstract

Recently, in-network aggregation to scale distributed machine learning (ML) has been presented. A network switch implementation uses IPv4 broadcast messages from switch to the hosts to send updates to all workers. IPv6 does not support broadcast addresses. This document proposes, IPv6 implementations use the IPv6 link-local all-nodes multicast address, until a new IPv6 link-local multicast address is assigned by IANA for switch to hosts multicast communications.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 11, 2020.

Table of Contents



## 1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

## 2. Introduction

New computing in the network for ML uses IPv4 broadcast communications between switch to hosts. [Switch-ML]. With IPv6, multicast communications would be used. This document proposes a new well-known multicast address be defined for such communications. Until a new address is defined, the IPv6 link-local all-nodes multicast address may be used. By definition, a layer-2 switch operates in the link-local subnet. Thus, the IPv6 link-local multicast address defined by this document suffices for switch to host multicast communications.

## 3. Additional Information

It is common when new networking protocols such as RPL [RFC6550] and Babel [RFC6126] were developed, each protocol requested IANA for a new IPv6 link-local multicast address for use. Switch ML does not have a protocol defined by IETF just yet and may never define one. However, experiments in switch ML are already using IP or layer-2 broadcast communications. For IPv6, switch ML experiments should use IPv6 link-local multicast communications. A new IPv6 link-local multicast address for switch ML facilitates efficient filtering by hosts.

If a switch is configured in layer-3 mode and if switch ML communicates with hosts to another IPv6 subnet, an IPv6 Site-Local Scope Multicast address is recommended for communications.

## 4. P4 Considerations

P4 issues arise when implementing the paper. A P4 header stack (array) is used to represent the vector. P4 has no for loops, so how does one add elements of the vector? There is the one way. You can have one action that adds 1 header, another that adds 2, another that adds 3, etc., and choose between them at run time via a table lookup. Even with a vector with 20-50 elements, writing such P4 code manually is risky. One has to write a Python script to generate such code. The paper did not think of it, but an alternative exists. With every element in the vector, use a new 1-byte field. The field has values 0 or 1. 1 represents bos (bottom of stack) like what MPLS uses. The last element has bos set to 1. Then, one has a termination condition in the P4 parser to terminate recursive parsing of vector. As the parser is recursing through the vector, add the vector elements and save sum in metadata. One does not use a bit for bos because the p4c bmv2 model accepts a header on byte boundary.

The same p4c bmv2 model, also, does not accept an index into array if the index is a run-time value. The p4-16 specification allows run-time index. Algorithm 1 in the paper uses "p.idx" which is a run-time value. Again, one has to use P4 tables. Include the run-time variable index as a table search key, which selects between several different actions that are identical, except for the constant array index values they use. This gets cumbersome with 128 or 512 indices in the paper. Again, one would have to generate P4 code to avoid mistakes cut-and-pasting code. For 128 indices with some bit shifting, maybe we reduces indices to 32. If an asic allows run-time index into array, then it is better to use the asic simulator rather than bmv2 simulator.

## 5. Security Considerations

Use IPSec [RFC4301].

## 6. IANA Considerations

This document requests IANA to assign a new IPv6 link-local multicast address for use by network ML. This multicast address name is Switch_ML_Host. An interface on the host MUST join this well-known multicast address.

Additionally, IANA is requested to assign a new IPv6 Site-Local Scope Multicast address for switch ML to host communication across IPv6

subnets. If configured to do so, an interface on the host MUST join this well-known multicast address.

## 7. Acknowledgements

Thanks (in alphabetical order by first name) to Marco Canini for his review.

## 8. References

### 8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC6126] Chroboczek, J., "The Babel Routing Protocol", RFC 6126, DOI 10.17487/RFC6126, April 2011, <https://www.rfc-editor.org/info/rfc6126>.

[RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <https://www.rfc-editor.org/info/rfc6550>.

### 8.2. Informative References

[Switch-ML] Sapio, A., Canini, M., Ho, C., Nelson, J., Kalnis, P., Kim, C., Krishnamurthy, A., Moshref, M., Ports, D. R., and P. Richtarik, "SwitchML: Scaling Distributed Machine Learning with In-Network Aggregation", February 2019, <https://arxiv.org/pdf/1903.06701.pdf>.

Author's Address

Hemant Singh
MNK Consulting
7 Caldwell Drive
Westford, MA 01886
USA

Phone: +1 978 692 2340
Email: hemant@mnkcg.com
URI: http://mnkcg.com/