Abstract

This document describes a generic TLV attribute encoding format to be added to PIM join messages.
Table of Contents

1. Conventions used in this document .................. 3
2. Introduction ........................................ 3
3. Use of the Attribute Field in Join Messages ........... 3
   3.1. Attribute join .................................. 3
   3.2. Transitive attributes ........................... 3
   3.3. Attribute Hello Option ......................... 3
   3.4. Conflicting attributes ......................... 4
   3.5. Attribute Convergence ........................... 5
   3.6. Multiple attributes ............................ 5
   3.7. Applicability of the attributes ................. 5
   3.8. PIM attribute packet format ...................... 5
      3.8.1. PIM Join packet format ..................... 5
      3.8.2. PIM Attribute Hello option ................. 6
4. IANA Considerations ................................ 6
5. Security Considerations ............................. 6
6. Acknowledgments ..................................... 7
7. References .......................................... 7
   7.1. Normative References ............................ 7
   7.2. Informative References ......................... 7
Authors’ Addresses ..................................... 7
Intellectual Property and Copyright Statements .......... 9
1. Conventions used in this document

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 and indicate requirement levels for compliant PIM-SM implementations.

2. Introduction

It is sometimes convenient to add additional information to PIM join messages. The generic PIM encoding format is not always optimal to do this. This document defines a new field in the PIM Join message that allows it to use TLVs, hereby called the attribute field. The content and purpose of this attribute field is outside the scope of this document, only the generic encoding format is described here.

3. Use of the Attribute Field in Join Messages

3.1. Attribute join

Attribute fields are defined similar to the PIM source encoding type as defined in [RFC4601]. A source address without any additional TLV’s should be processed identically to a source address in the default source encoding.

Multiple TLV’s from the same or different type are permitted in a single source address in any order.

3.2. Transitive attributes

It may be desired to have routers that understand the generic attribute format, forward the attributes regardless of whether they understand the TLV’s encoded in the attribute not. For this the first bit in the Type field is reserved. If this bit is set then the router MUST forward the TLV upstream in case the router does not understand that type. If this bit is not set the router MUST NOT forward the TLV upstream in the case the router does not understand that type.

3.3. Attribute Hello Option

A new PIM source type has been defined to include the Attribute field. This source type is included in a normal PIM Join. Each router on a connected network needs to be able to understand and parse the Join message. Therefore we include a new PIM hello option to advertise our capability to parse and process the new source type.
We can only send a PIM Join which includes an attribute if ALL routers on the network support the new option. (Even a router which is not the upstream neighbor must be able parse to the packet in order to do Join suppression or overriding.) Option value TBD.

Having the attribute hello option does not guarantee that all neighbors understand all possible individual attributes. As there’s no immediate way to act on a neighbor’s incapability to process certain attribute types, it is not desired to have a hello option for each possible attribute type.

3.4. Conflicting attributes

It’s possible that a router receives conflicting attribute information from different downstream routers. Conflicts only occur with attributes of the same type. If two different attributes of two different types are received they should both be processed and forwarded.

```
       ( Edge A1 )   ( Edge B1 )---- [R1]
/     /\       /\            /
/    /   \    /   \          /   /
  [S]  /     \  ( Core )------ /
/     /   \                      /   \
\    /     \                       /   \
( Edge A2 )   ( Edge B2 )---- [R2]
```

Figure 2

An example join attribute in this case is an exit router. There are 2 receivers for the same group connected to Edge B1 and B2. Suppose that edge router B1 prefers A1 as the exit point and B2 prefers A2 as exit point to reach the source S. If both Edge B1 and B2 send a Join including an attribute to prefer their exit router in the network and they cross the same core router, the core router will get conflicting attribute information for the source. If this happens we use the Attribute from the PIM adjacency with the numerically smallest IP address. In the case of IPv6, the link local address will be used. When two neighbors have the same IP address, either for IPv4 or IPv6, the interface index must be used as a tie breaker. The attributes from other sending routers may be kept around in case the best attribute gets pruned or expires, we are able to immediately use the second best attribute and converge quickly without waiting for the next periodic update. If a TLV has its own definition for conflict resolution it is preferred over the conflict resolution above.
3.5. Attribute Convergence

An attribute is included in a PIM Join message together with the source information. If the attribute for this source is changed, we trigger a new PIM Join message to the upstream router. This causes the new attribute to be propagated. This new attribute implicitly removes the old attribute upstream. If processing the new attribute results in a change in the distribution tree, a PIM Prune message may be sent. This PIM Prune does not need to carry any attribute, the sender of the prune and the source and group information is enough to identify the entry. The attribute information is removed immediately and possibly a new attribute is chosen from the database if available.

3.6. Multiple attributes

A PIM Join can contain multiple attributes. The attributes are encoded as TLVs associated with a new PIM source type in the PIM message. When a PIM Join with multiple attributes is received, each type is processed separately. For each type, the first attribute of that type is processed, and the action taken depends upon the type. This may or may not result in the processing of the next attribute. Attributes that the router understands but are not processed MUST be passed upstream unchanged.

3.7. Applicability of the attributes

PIM Joins with attributes can be applied to both shared-trees rooted at a Rendezvous Point (RP) and shortest-path trees as described in [RFC4601].

3.8. PIM attribute packet format

3.8.1. PIM Join packet format

There is no space in the default PIM source encoding to include a attribute field. Therefore we introduce a new source encoding type. The attributes are formatted as TLV’s. The new Encoded source address looks like this:
F bit, Forward Unknown TLV. If this bit is set the TLV is forwarded regardless if the router understands the Type.

S bit, Bottom of Stack. If this bit is set then this is the last TLV in the stack.

Type field of the TLV is 6 bits.

Length field of the TLV is 1 byte.

The other fields are the same as described in the RFC 4601.

The source TLV encoding type: TBD.

3.8.2. PIM Attribute Hello option

Option type: TBD.

4. IANA Considerations

This document requires a new PIM Hello Option value and a new PIM Join Encoding type.

5. Security Considerations

Security of the join attribute is only guaranteed by the security of
the PIM packet, so the security considerations for PIM join packets as described in PIM-SM [RFC4601] apply here.

6. Acknowledgments

The authors would like to thank Stig Venaas, James Lingard, Bharat Joshi, Marshall Eubanks, Pekka Savola and Tom Pusateri for their input.

7. References

7.1. Normative References


7.2. Informative References

Authors’ Addresses

Arjen Boers
Cisco Systems, Inc.
Avda. Diagonal, 682
Barcelona 08034
Spain

Email: aboers@cisco.com

IJJsbrand Wijnands
Cisco Systems, Inc.
De kleetlaan 6a
Diegem 1831
Belgium

Email: ice@cisco.com
Eric Rosen
Cisco Systems, Inc.
1414 Massachusetts Avenue
Boxborough, Ma  01719

Email: erosen@cisco.com
Full Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).