

INTERNET-DRAFT  
[draft-ietf-msdp-spec-20.txt](#)  
Category  
Expires: November 2003

Bill Fenner (Editor)  
David Meyer (Editor)  
Experimental  
May 2003

**Multicast Source Discovery Protocol (MSDP)**

**<[draft-ietf-msdp-spec-20.txt](#)>**

Status of this Document

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This document is a product of an individual. Comments are solicited and should be addressed to the author(s).

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

### Abstract

The Multicast Source Discovery Protocol, MSDP, describes a mechanism to connect multiple IP Version 4 Protocol Independent Multicast Sparse-Mode (PIM-SM) [[RFC2362](#)] domains together. Each PIM-SM domain uses its own independent Rendezvous Point (RP) and does not have to depend on RPs in other domains. This document reflects existing MSDP implementations.

## Table of Contents

<a href="#">1. Introduction</a>	<a href="#">5</a>
<a href="#">2. Overview</a>	<a href="#">5</a>
<a href="#">3. Procedure</a>	<a href="#">5</a>
<a href="#">4. Caching</a>	<a href="#">7</a>
<a href="#">5. Timers</a>	<a href="#">7</a>
<a href="#">5.1. SA-Advertisement-Timer</a>	<a href="#">7</a>
<a href="#">5.2. SA-Advertisement-Timer Processing</a>	<a href="#">8</a>
<a href="#">5.3. SA Cache Timeout (SA-State Timer)</a>	<a href="#">8</a>
<a href="#">5.4. Peer Hold Timer</a>	<a href="#">8</a>
<a href="#">5.5. KeepAlive Timer</a>	<a href="#">9</a>
<a href="#">5.6. ConnectRetry Timer</a>	<a href="#">9</a>
<a href="#">6. Intermediate MSDP Peers</a>	<a href="#">9</a>
<a href="#">7. SA Filtering and Policy</a>	<a href="#">10</a>
<a href="#">8. Encapsulated Data Packets</a>	<a href="#">10</a>
<a href="#">9. Other Scenarios</a>	<a href="#">10</a>
<a href="#">10. MSDP Peer-RPF Forwarding</a>	<a href="#">11</a>
<a href="#">10.1. Definitions</a>	<a href="#">11</a>
<a href="#">10.1.1. Multicast RPF Routing Information Base</a>	<a href="#">11</a>
<a href="#">10.1.2. Peer-RPF Route</a>	<a href="#">11</a>
<a href="#">10.1.3. Peer-RPF Forwarding Rules</a>	<a href="#">11</a>
<a href="#">10.2. MSDP mesh-group semantics</a>	<a href="#">13</a>
<a href="#">11. MSDP Connection State Machine</a>	<a href="#">14</a>
<a href="#">11.1. Events</a>	<a href="#">15</a>
<a href="#">11.2. Actions</a>	<a href="#">16</a>
<a href="#">11.3. Peer-specific Events</a>	<a href="#">16</a>
<a href="#">11.4. Peer-independent Events</a>	<a href="#">17</a>
<a href="#">12. Packet Formats</a>	<a href="#">17</a>
<a href="#">12.1. MSDP TLV format</a>	<a href="#">17</a>
<a href="#">12.2. Defined TLVs</a>	<a href="#">18</a>
<a href="#">12.2.1. IPv4 Source-Active TLV</a>	<a href="#">18</a>
<a href="#">12.2.2. KeepAlive TLV</a>	<a href="#">20</a>
<a href="#">13. MSDP Error Handling</a>	<a href="#">20</a>
<a href="#">14. SA Data Encapsulation</a>	<a href="#">21</a>
<a href="#">15. Applicability Statement</a>	<a href="#">21</a>
<a href="#">15.1. Between PIM Domains</a>	<a href="#">21</a>
<a href="#">15.2. Between Anycast-RPs</a>	<a href="#">21</a>
<a href="#">16. Intellectual Property</a>	<a href="#">21</a>
<a href="#">17. Acknowledgments</a>	<a href="#">22</a>
<a href="#">18. Security Considerations</a>	<a href="#">23</a>
<a href="#">19. IANA Considerations</a>	<a href="#">23</a>
<a href="#">19.1. IANA Allocated TLV Range</a>	<a href="#">23</a>
<a href="#">19.2. Experimental TLV Range</a>	<a href="#">23</a>
<a href="#">20. References</a>	<a href="#">24</a>
<a href="#">20.1. Normative References</a>	<a href="#">24</a>
<a href="#">20.2. Informative References</a>	<a href="#">24</a>



<a href="#">21</a> . Editor's Addresses. . . . .	<a href="#">25</a>
<a href="#">22</a> . Full Copyright Statement. . . . .	<a href="#">25</a>

## **1. Introduction**

The Multicast Source Discovery Protocol, MSDP, describes a mechanism to connect multiple PIM Sparse-Mode (PIM-SM) [[RFC2362](#)] domains together. Each PIM-SM domain uses its own independent RP(s) and does not have to depend on RPs in other domains. Advantages of this approach include:

- o No Third-party resource dependencies on a domain's RP

PIM-SM domains can rely on their own RPs only.

- o Receiver only Domains

Domains with only receivers get data without globally advertising group membership.

Note that MSDP may be used with protocols other than PIM-SM, but such usage is not specified in this memo.

## **2. Overview**

MSDP-speaking routers in a PIM-SM domain have a MSDP peering relationship with MSDP peers in another domain. The peering relationship is made up of a TCP connection in which control information is exchanged. Each domain has one or more connections to this virtual topology.

The purpose of this topology is to allow domains to discover multicast sources from other domains. If the multicast sources are of interest to a domain which has receivers, the normal source-tree building mechanism in PIM-SM will be used to deliver multicast data over an inter-domain distribution tree.

## **3. Procedure**

When an RP in a PIM-SM domain first learns of a new sender, e.g. via PIM register messages, it constructs a "Source-Active" (SA) message and sends it to its MSDP peers. All RPs, which intend to originate or



receive SA messages, must establish MSDP peering with other RPs, either directly or via an intermediate MSDP peer. The SA message contains the following fields:

- o Source address of the data source.
- o Group address the data source sends to.
- o IP address of the RP.

Note that an RP that isn't a DR on a shared network SHOULD NOT originate SA's for directly connected sources on that shared network; it should only originate in response to receiving Register messages from the DR.

Each MSDP peer receives and forwards the message away from the RP address in a "peer-RPF flooding" fashion. The notion of peer-RPF flooding is with respect to forwarding SA messages. The Multicast RPF Routing Information Base (MRIB) is examined to determine which peer towards the originating RP of the SA message is selected. Such a peer is called an "RPF peer". See [section 13](#) for the details of peer-RPF forwarding.

If the MSDP peer receives the SA from a non-RPF peer towards the originating RP, it will drop the message. Otherwise, it forwards the message to all its MSDP peers (except the one from which it received the SA message).

When an MSDP peer which is also an RP for its own domain receives a new SA message, it determines if there are any group members within the domain interested in any group described by an (Source, Group), or (S,G) entry within the SA message. That is, the RP checks for a (\*,G) entry with a non-empty outgoing interface list; this implies that some system in the domain is interested in the group. In this case, the RP triggers a (S,G) join event towards the data source as if a Join/Prune message was received addressed to the RP itself. This sets up a branch of the source-tree to this domain. Subsequent data packets arrive at the RP via this tree branch, and are forwarded down the shared-tree inside the domain. If leaf routers choose to join the source-tree they have the option to do so according to existing PIM-SM conventions. Finally, if an RP in a domain receives a PIM Join message for a new group G, the RP SHOULD trigger a (S,G) join event for each active (S,G) for that group in its SA cache.

This procedure has been affectionately named flood-and-join because if any RP is not interested in the group, they can ignore the SA message. Otherwise, they join a distribution tree.





#### **4. Caching**

A MSDP speaker MUST cache SA messages. Caching allows pacing of MSDP messages as well as reducing join latency for new receivers of a group G at an originating RP which has existing MSDP (S,G) state. In addition, caching greatly aids in diagnosis and debugging of various problems.

An MSDP speaker must provide a mechanism to reduce the forwarding of new SA's. The SA-cache is used to reduce storms and performs this by not forwarding SA's unless they are in the cache or are new SA packets that the MSDP speaker will cache for the first time. The SA-cache also reduces storms by advertising from the cache at a period of no more than twice per SA-Advertisement-Timer interval and not less than 1 time per SA Advertisement period.

#### **5. Timers**

The main timers for MSDP are: SA-Advertisement-Timer, SA Cache Entry timer, Peer Hold Timer, KeepAlive timer, and ConnectRetry timer. Each is considered below.

##### **5.1. SA-Advertisement-Timer**

RPs which originate SA messages do so periodically as long as there is data being sent by the source. There is one SA-Advertisement-Timer covering the sources that an RP may advertise. [SA-Advertisement-Period] MUST be 60 seconds. An RP MUST not send more than one periodic SA message for a given (S,G) within an SA Advertisement interval. Originating periodic SA messages is required to keep announcements alive in caches. Finally, an originating RP SHOULD trigger the transmission of an SA message as soon as it receives data from an internal source for the first time. This initial SA message may be in addition to the periodic sa-message forwarded in that first 60 seconds for that (S,G).



## **5.2. SA-Advertisement-Timer Processing**

An RP MUST spread the generation of periodic SA messages (i.e. messages advertising the active sources for which it is the RP) over its reporting interval (i.e. SA-Advertisement-Period). An RP starts the SA-Advertisement-Timer when the MSDP process is configured. When the timer expires, an RP resets the timer to [SA-Advertisement-Period] seconds, and begins the advertisement of its active sources. Active sources are advertised in the following manner: An RP packs its active sources into an SA message until the largest MSDP packet that can be sent is built or there are no more sources, and then sends the message. This process is repeated periodically within the SA-Advertisement-Period in such a way that all of the RP's sources are advertised. Note that since MSDP is a periodic protocol, an implementation SHOULD send all cached SA messages when a connection is established. Finally, the timer is deleted when the MSDP process is de-configured.

## **5.3. SA Cache Timeout (SA-State Timer)**

Each entry in an SA Cache has an associated SA-State Timer. A (S,G)-SA-State-Timer is started when an (S,G)-SA message is initially received by an MSDP peer. The timer is reset to [SG-State-Period] if another (S,G)-SA message is received before the (S,G)-SA-State Timer expires. [SG-State-Period] MUST NOT be less than [SA-Advertisement-Period] + [SA-Hold-Down-Period].

## **5.4. Peer Hold Timer**

The Hold Timer is initialized to [HoldTime-Period] when the peer's transport connection is established, and is reset to [HoldTime-Period] when any MSDP message is received. Finally, the timer is deleted when the peer's transport connection is closed. [HoldTime-Period] MUST be at least three seconds. The recommended value for [HoldTime-Period] is 75 seconds.



### **5.5. KeepAlive Timer**

Once an MSDP transport connection is established, each side of the connection sends a KeepAlive message and sets a KeepAlive timer. If the KeepAlive timer expires, the local system sends a KeepAlive message and restarts its KeepAlive timer.

The KeepAlive timer is set to [[KeepAlive-Period](#)] when the peer comes up. The timer is reset to [[KeepAlive-Period](#)] each time an MSDP message is sent to the peer, and reset when the timer expires.

Finally, the KeepAlive timer is deleted when the peer's transport connection is closed.

[KeepAlive-Period] MUST be less than [HoldTime-Period], and MUST be at least one second. The recommended value for [[KeepAlive-Period](#)] is 60 seconds.

### **5.6. ConnectRetry Timer**

The ConnectRetry timer is used by the MSDP peer with the lower IP address to transition from INACTIVE to CONNECTING states. There is one timer per peer, and the [ConnectRetry-Period] SHOULD be set to 30 seconds. The timer is initialized to [ConnectRetry-Period] when an MSDP speaker attempts to actively open a TCP connection to its peer (see [section 15](#), event E2, action A2 ). When the timer expires, the peer retries the connection and the timer is reset to [ConnectRetry-Period]. It is deleted if either the connection transitions into ESTABLISHED state or the peer is de-configured.

## **6. Intermediate MSDP Peers**

Intermediate MSDP speakers do not originate periodic SA messages on behalf of sources in other domains. In general, an RP MUST only originate an SA for a source which would register to it, and ONLY RPs may originate SA messages. Intermediate MSDP speakers MAY forward SA messages received from other domains.



## **7. SA Filtering and Policy**

As the number of (S,G) pairs increases in the Internet, an RP may want to filter which sources it describes in SA messages. Also, filtering may be used as a matter of policy which at the same time can reduce state. MSDP peers in transit domains should not filter SA messages or the flood-and-join model can not guarantee that sources will be known throughout the Internet (i.e., SA filtering by transit domains may cause undesired lack of connectivity). In general, policy should be expressed using MBGP [[RFC2283](#)]. This will cause MSDP messages to flow in the desired direction and peer-RPF fail otherwise. An exception occurs at an administrative scope [[RFC2365](#)] boundary. In particular, a SA message for a (S,G) MUST NOT be sent to peers which are on the other side of an administrative scope boundary for G.

## **8. Encapsulated Data Packets**

The RP MAY encapsulate multicast data from the source. An interested RP may decapsulate the packet, which SHOULD be forwarded as if a PIM register encapsulated packet was received. That is, if packets are already arriving over the interface toward the source, then the packet is dropped. Otherwise, if the outgoing interface list is non-null, the packet is forwarded appropriately. Note that when doing data encapsulation, an implementation MUST bound the time during which packets are encapsulated.

This allows for small bursts to be received before the multicast tree is built back toward the source's domain. For example, an implementation SHOULD encapsulate at least the first packet to provide service to bursty sources.

## **9. Other Scenarios**

MSDP is not limited to deployment across different routing domains. It can be used within a routing domain when it is desired to deploy multiple RPs for the same group ranges such as with Anycast RP's. As long as all RPs have a interconnected MSDP topology, each can learn about active sources as well as RPs in other domains.





## **10. MSDP Peer-RPF Forwarding**

The MSDP Peer-RPF Forwarding rules are used for forwarding SA messages throughout an MSDP enabled internet. Unlike the RPF check used when forwarding data packets, which generally compares the packet's source address against the interface upon which the packet was received, the Peer-RPF check compares the RP address carried in the SA message against the MSDP peer from which the message was received.

### **10.1. Definitions**

The following definitions are used in the description of the Peer-RPF Forwarding Rules:

#### **10.1.1. Multicast RPF Routing Information Base**

The Multicast RPF Routing Information Base (MRIB) is the multicast topology table. It is typically derived from the unicast routing table or from other routing protocols such as multi-protocol BGP [[RFC2283](#)].

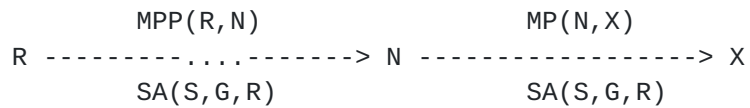
#### **10.1.2. Peer-RPF Route**

The Peer-RPF route is the route that the MRIB chooses for a given address. The Peer-RPF route for a SA's originating RP is used to select the peer from which the SA is accepted.

#### **10.1.3. Peer-RPF Forwarding Rules**

An SA message originated by R and received by X from N is accepted if N is the peer-RPF neighbor for X, and is discarded otherwise.





MP(N,X) is an MSDP peering between N and X. MPP(R,N) is an MSDP peering path (zero or more MSDP peers) between R and N, e.g. MPP(R,N) = MP(R, A) + MP(A, B) + MP(B, N). SA(S,G,R) is an SA message for source S on group G originated by an RP R.

The peer-RPF neighbor N is chosen deterministically, using the first of the following rules that matches. In particular, N is the RPF neighbor of X with respect to R if



- (i). N == R (X has an MSDP peering with R).
- (ii). N is the eBGP NEXT\_HOP of the Peer-RPF route for R.
- (iii). The Peer-RPF route for R is learned through a distance-vector or path-vector routing protocol (e.g. BGP, RIP, DVMRP) and N is the neighbor that advertised the Peer-RPF route for R (e.g. N is the iBGP advertiser of the route for R), or N is the IGP next hop for R if the route for R is learned via a link-state protocol (e.g. OSPF [[RFC2178](#)] or IS-IS [[RFC1142](#)]).
- (iv). N resides in the closest AS in the best path towards R. If multiple MSDP peers reside in the closest AS, the peer with the highest IP address is the rpf-peer.
- (v). N is configured as the static RPF-peer for R.

MSDP peers, which are NOT in state ESTABLISHED (i.e., down peers), are not eligible for peer RPF consideration.

## **10.2. MSDP mesh-group semantics**

An MSDP mesh-group is a operational mechanism for reducing SA flooding, typically in an intra-domain setting. In particular, when some subset of a domain's MSDP speakers are fully meshed, they can be configured into a mesh-group.

Note that mesh-groups assume that a member doesn't have to forward an SA to other members of the mesh-group because the originator will forward to all members. To be able for the originator to forward to all members (and to have each member also be a potential originator), the mesh-group must be a full mesh of MSDP peering among all members.

The semantics of the mesh-group are as follows:

- (i). If a member R of a mesh-group M receives a SA message from an MSDP peer that is also a member of mesh-group M, R accepts the SA message and forwards it to all of its peers that are not part of mesh-group M. R MUST NOT forward the SA message to other members of mesh-group M.



- (ii). If a member R of a mesh-group M receives an SA message from an MSDP peer that is not a member of mesh-group M, and the SA message passes the peer-RPF check, then R forwards the SA message to all members of mesh-group M and to any other msdp peers.

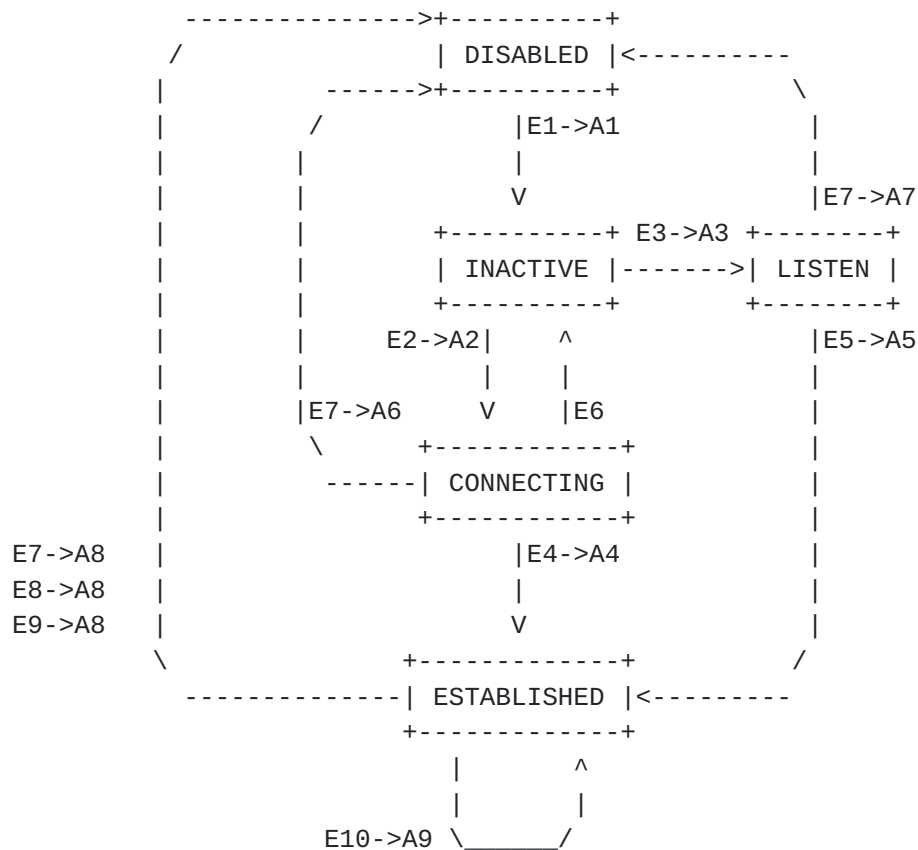
## **11. MSDP Connection State Machine**

MSDP uses TCP as its transport protocol. In a peering relationship, one MSDP peer listens for new TCP connections on the well-known port 639. The other side makes an active connect to this port. The peer with the higher IP address will listen. This connection establishment algorithm avoids call collision. Therefore, there is no need for a call collision procedure. It should be noted, however, that the disadvantage of this approach is that the startup time depends completely upon the active side and its connect retry timer; the passive side cannot cause the connection to be established.

An MSDP peer starts in the DISABLED state. MSDP peers establish peering sessions according to the following state machine:







### 11.1. Events

- E1) Enable MSDP peering with P
- E2) Own IP address < P's IP address
- E3) Own IP address > P's IP address
- E4) TCP established (active side)
- E5) TCP established (passive side)
- E6) ConnectRetry timer expired
- E7) Disable MSDP peering with P (e.g. when one's own address is changed)
- E8) Hold Timer expired
- E9) MSDP TLV format error detected
- E10) Any other error detected



### **11.2. Actions**

- A1) Allocate resources for peering with P Compare one's own and peer's IP addresses
- A2) TCP active OPEN Set ConnectRetry timer to [ConnectRetry-Period]
- A3) TCP passive OPEN (listen)
- A4) Delete ConnectRetry timer Send KeepAlive TLV  
Set KeepAlive timer to [[KeepAlive-Period](#)]  
Set Hold Timer to [HoldTime-Period]
- A5) Send KeepAlive TLV  
Set KeepAlive timer to [[KeepAlive-Period](#)]  
Set Hold Timer to [HoldTime-Period]
- A6) Abort TCP active OPEN attempt  
Release resources allocated for peering with P
- A7) Abort TCP passive OPEN attempt  
Release resources allocated for peering with P
- A8) Close the TCP connection  
Release resources allocated for peering with P
- A9) Drop the packet

### **11.3. Peer-specific Events**

The following peer-specific events can occur in the ESTABLISHED state, they do not cause a state transition. Appropriate actions are listed for each event.

- \*) KeepAlive timer expired:
  - > Send KeepAlive TLV
  - > Set KeepAlive timer to [[KeepAlive-Period](#)]
- \*) KeepAlive TLV received:
  - > Set Hold Timer to [HoldTime-Period]
- \*) Source-Active TLV received:
  - > Set Hold Timer to [HoldTime-Period]
  - > Run Peer-RPF Forwarding algorithm
  - > Set KeepAlive timer to [[KeepAlive-Period](#)] for those peers the Source-Active TLV is forwarded to
  - > Send information to PIM-SM
  - > Store information in cache



#### 11.4. Peer-independent Events

There are also a number of events that affect more than one peering session, but still require actions to be performed on a per-peer basis.

- \* ) SA-Advertisement-Timer expired:
  - > Start periodic transmission of Source-Active TLV(s)
  - > Set KeepAlive timer to [[KeepAlive-Period](#)] each time a Source-Active TLV is sent
- \* ) MSDP learns of a new active internal source (e.g. PIM-SM register received for a new source):
  - > Send Source-Active TLV
  - > Set KeepAlive timer to [[KeepAlive-Period](#)]
- \* ) SG-State-Timer expired (one timer per cache entry):
  - > Implementation specific, typically mark the cache entry for deletion

## 12. Packet Formats

MSDP messages are encoded in TLV format. If an implementation receives a TLV whose length exceeds the maximum TLV length specified below, the TLV SHOULD be accepted. Any additional data, including possible next TLV's in the same message, SHOULD be ignored, and the MSDP session should not be reset.

### 12.1. MSDP TLV format

```

      0                               1                               2                               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|   Type   |                               Length                               | Value .... |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

Type (8 bits)

Describes the format of the Value field.

Length (16 bits)

Length of Type, Length, and Value fields in octets.



Minimum length required is 4 octets, except for Keepalive messages. The maximum TLV length is 9192.

Value (variable length)

Format is based on the Type value. See below. The length of the value field is Length field minus 3. All reserved fields in the Value field MUST be transmitted as zeros and ignored on receipt.

## **12.2. Defined TLVs**

The following TLV Types are defined:

Code	Type
=====	
1	IPv4 Source-Active
2	IPv4 Source-Active Request
3	IPv4 Source-Active Response
4	KeepAlive
5	Reserved (Previously: Notification)

Each TLV is described below.

In addition, the following TLV Types are assigned but not described in this memo:

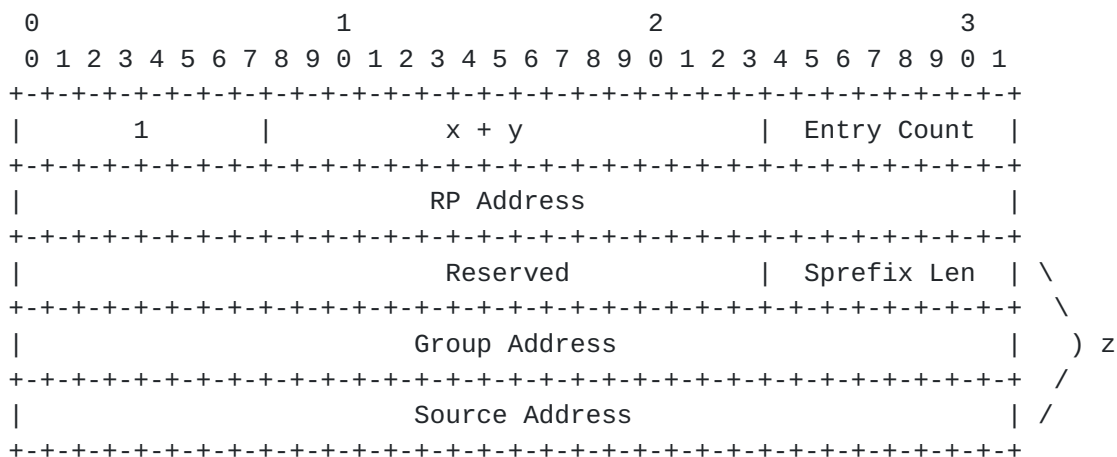
Code	Type
=====	
6	MSDP traceroute in progress
7	MSDP traceroute reply

### **12.2.1. IPv4 Source-Active TLV**

The maximum size SA message that can be sent is 9192 octets. The 9192 octet size does not include the TCP, IP, layer-2 headers.







## Type

IPv4 Source-Active TLV is type 1.

Length x

Is the length of the control information in the message. x is 8 octets (for the first two 32-bit quantities) plus 12 times Entry Count octets.

Length  $y$

If 0, then there is no data encapsulated. Otherwise an IPv4 packet follows and y is the value of the total length field in the header of the encapsulated IP packet. If there are multiple (S,G) entries in an SA message, only the last entry may have encapsulated data and it must reflect the source and destination addresses in the header of the encapsulated IP packet.

## Entry Count

Is the count of z entries (note above) which follow the RP address field. This is so multiple (S,G)s from the same domain can be encoded efficiently for the same RP address. An SA message containing encapsulated data typically has an entry count of 1 (i.e. only contains a single entry, for the (S,G) representing the encapsulated packet).

RP Address

The address of the RP in the domain the source has become active in.

Reserved

The Reserved field MUST be transmitted as zeros and MUST be ignored by a receiver.

Sprefix Len



If an MSDP message is received with a TLV format error, the session SHOULD be reset with that peer. MSDP messages with other errors, such as unrecognized type code, received from MSDP peers, SHOULD be silently discarded and the session SHOULD not be reset.



## **14. SA Data Encapsulation**

As discussed earlier, TCP encapsulation of data in SA messages MAY be supported for backwards compatibility with legacy MSDP peers.

## **15. Applicability Statement**

MSDP is used primarily in two deployment scenarios:

### **15.1. Between PIM Domains**

MSDP can be used between PIM domains to convey information about active sources available in other domains. MSDP peering used in such cases is generally one to one peering, and utilizes the deterministic peer-RPF rules described in this spec (i.e., does not use mesh-groups). Peerings can be aggregated on a single MSDP peer, typically from one to hundreds of peerings, similar in scale, although not necessarily consistent, with BGP peerings.

### **15.2. Between Anycast-RPs**

MSDP is also used between Anycast-RPs [[RFC3446](#)] within a PIM domain to synchronize information about the active sources being served by each Anycast-RP peer (by virtue of IGP reachability). MSDP peering used in this scenario is typically based on MSDP mesh groups, where anywhere from two to tens of peers can comprise a given mesh group, although more than ten is not typical. One or more of these mesh-group peers may then also have additional one-to-one peering with msdp peers outside that PIM domain as described in scenario A, for discovery of external sources. MSDP for anycast-RP without external MSDP peering is a valid deployment option and common.

## **16. Intellectual Property**

The IETF takes no position regarding the validity or scope of any intellectual property 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; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in [BCP-11](#). Copies of claims of rights made available for publication 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 implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

## **[17.](#) Acknowledgments**

The editors would like to thank the original authors, Dino Farinacci, Yakov Rehkter, Peter Lothberg, Hank Kilmer, and Jerney Hall for their original contribution to the MSDP specification. In addition, Bill Nickless, John Meylor, Liming Wei, Manoj Leelanivas, Mark Turner, John Zwiebel, Cristina Radulescu-Banu, Brian Edwards, Selina Priestley, IJsbrand Wijnands, Tom Pusateri, Kristofer Warell, Henning Eriksson, Thomas Eriksson, Dave Thaler, and Ravi Shekhar provided useful and productive design feedback and comments. Toerless Eckert, Leonard Giuliano, Mike McBride, David Meyer, John Meylor, Pekka Savola, Ishan Wu, and Swapna Yelamanchi contributed to the final version of the draft.





## **18. Security Considerations**

An MSDP implementation MUST implement Keyed MD5 [[RFC2385](#)] to secure control messages, and MUST be capable of interoperating with peers that do not support it. However, if one side of the connection is configured with Keyed MD5 and the other side is not, the connection SHOULD NOT be established.

In addition, to mitigate state explosion during denial of service and other attacks, SA filters and limits SHOULD be used with MSDP to limit the sources and groups that will be passed between RPs [[DEPLOY](#)]. These filtering and limiting functions may include, for example, access lists of source or group addresses which should not be propagated to other domains using MSDP, the absolute highest acceptable number of SA-state entries or a rate-limit of for the creation of new SA-state entries after the connection has been established.

## **19. IANA Considerations**

This document creates a new namespace called "MSDP TLV Values" that the IANA should manage. The initial seven MSDP TLV values are specified in [Section 12.2](#). The following two sections describe the rules for allocating new MSDP TLV values.

### **19.1. IANA Allocated TLV Range**

MSDP TLV values in the range [8,200] (inclusive) are to be allocated using an IESG Approval or Standards Action process [[RFC2434](#)].

### **19.2. Experimental TLV Range**

TLV values in the range [201,255] (inclusive) are allocated for experimental use.



## [20.](#) References

### [20.1.](#) Normative References

- [RFC1142] Oran, D. "OSI IS-IS Intra-domain Routing Protocol", [RFC 1142](#), February 1990.
- [RFC2178] Moy, J., "OSPF Version 2", [RFC 2178](#), April, 1998.
- [RFC2283] Bates, T., R. Chandra, and D. Katz, and Y. Rekhter., "Multiprotocol Extensions for BGP-4", [RFC 2283](#), February 1998.
- [RFC2362] Estrin D., et al., "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification", [RFC 2362](#), June 1998.
- [RFC2365] Meyer, D. "Administratively Scoped IP Multicast", [RFC 2365](#), July, 1998.
- [RFC2385] Heffernan. A, "Protection of BGP Sessions via the TCP MD5 Signature Option", August, 1998.
- [RFC2434] Narten, T., and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", [RFC2434](#)/BCP0026, October, 1998.
- [RFC3446] Kim, D., et al., "Anycast Rendezvous Point (RP) Mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)", [RFC 3446](#), January, 2003.

### [20.2.](#) Informative References

- [DEPLOY] McBride, M., J. Meylor, and D. Meyer, "Multicast Source Discovery Protocol (MSDP) Deployment Scenarios", [draft-ietf-mboned-msdp-deploy-02.txt](#), Work in Progress.
- [RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](#), March, 1997.



## **21. Editor's Addresses**

Bill Fenner  
AT&T Labs -- Research  
75 Willow Road  
Menlo Park, CA 94025  
Email: fenner@research.att.com

David Meyer  
Email: dmm@maoz.com

## **22. Full Copyright Statement**

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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.

