RFC Compliance Test Report

PIMV6 Results

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Type	FRR	FRR	FRR	FRR	FRR			
Commit ID	Released							
Commit Date	2022-11-03							
PIM-SMV6-1.1 **MAY**	draft-ietf-pim-sm-v2-new-12.txt s3. p8-9 PIM-SM Protocol Overview							
	`In phase one, a multicast receiver expresses its interest in receiving traffic destined for a multicast group.  Typically it does this using IGMP[6] or MLD[4], but other mechanisms might also serve this purpose.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-1.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s3. p8 PIM-SM Protocol Overview							
	`Regardless of how it is created, the primary role of the MRIB in the PIM protocol is to provide the next hop router along a multicast-capable path to each destination subnet.The MRIB is used to determine the next hop neighbor to which any PIM Join/Prune message is sent`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-1.3 **MUST**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s3. p8 PIM-SM Protocol Overview							
	`Regardless of how it is created, the primary role of the MRIB in the PIM protocol is to provide the next hop router along a multicast-capable path to each destination subnet. The MRIB is used to determine the next hop neighbor to which any PIM Join/Prune message is sent`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-1.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s3. p9 PIM-SM Protocol Overview							
	`Join messages are resent periodically so long as the receiver remains in the group`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-1.5 **MUST**	NEGATIVE: draft-ietf-pim-sm-v2-new-12.txt s3. p9 PIM-SM Protocol Overview							
	`The RP receives these encapsulated data packets, decapsulates them, and forwards them onto the shared tree.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-1.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s3 p9-10 PIM-SM Protocol Overview							
	`Although Register-encapsulation may continue indefinitely, for these reasons, the RP will normally choose to switch to native forwarding. To do this, when the RP receives a register-encapsulated data packet from source S on group G, it will normally initiate an (S,G) source-specific Join towards S.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-1.7 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s3 p10 PIM-SM Protocol Overview							
	`When packets from S also start to arrive natively at the RP, the RP will be receiving two copies of each of these packets. At this point, the RP starts to discard the encapsulated copy of these packets, and it sends a RegisterStop message back to S's DR to prevent the DR unnecessarily encapsulating the packets.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-1.8 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s3 p10 PIM-SM Protocol Overview							
	`To obtain lower latencies, a router on the receiver's LAN, typically the DR, may optionally initiate a transfer from the shared tree to a source-specific shortest-path tree (SPT). To do this, it issues an (S,G) Join towards S.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-1.9 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s3 p10-11 PIM-SM Protocol Overview							
	`At this point the receiver (or a router upstream of the receiver) will be receiving two copies of the data - one from the SPT and one from the RPT. When the first traffic starts to arrive from the SPT, the DR or upstream router starts to drop the packets for G from S that arrive via the RP tree.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-1.10 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s3 p11 PIM-SM Protocol Overview							
	`At this point the receiver (or a router upstream of the receiver) will be receiving two copies of the data - one from the SPT and one from the RPT. When the first traffic starts to arrive from the SPT, the DR or upstream router starts to drop the packets for G from S that arrive via the RP tree. In addition, it sends an (S,G) Prune message towards the RP. This is known as an (S,G,rpt) Prune. Here DUT is considered as an upstream router. The verification is made that the Join/Prune msg send by DUT has RPT-bit set`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-1.11 **MAY**	draft-ietf-pim-sm-v2-new-12.txt s3 p12 PIM-SM Protocol Overview							
	`PIM-SM routers need to know the address of the RP for each group for which they have (*,G) state. This address is obtained through a bootstrap mechanism.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-1.12 **MAY**	draft-ietf-pim-sm-v2-new-12.txt s3. p12 PIM-SM Protocol Overview							
	`PIM-SM routers need to know the address of the RP for each group for which they have (*,G) state. This address is obtained through static configuration.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-1.13 **MUST**	ANVL Setup Verification							
	Quick test to verify that DUT sends Assert message with metric value correctly							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-1.14 **MUST**	ANVL Setup Verification							
	Quick test to verify that DUT sends Assert message with metric preference value correctly							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-1.15 **MUST**	ANVL Setup Verification							
	Quick test to verify that DUT sends Register message with IP Source set to the IP address where it come from.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-3.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.3 p17 (*,G) State							
	The upstream (*,G) Join/Prune timer is used send out to override Prune(*,G) messages from peers on an upstream LAN interface							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-3.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.3 p17 (*,G) State							
	`The last RPF neighbor towards the RP is stored because if the MRIB changes then the RPF neighbor towards the RP may change. If it does so, then we need to trigger a new Join (*,G) to the new upstream neighbor`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-3.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.3 p17 (*,G) State							
	`The last RPF neighbor towards the RP is stored because if the MRIB changes then the RPF neighbor towards the RP may change. If it does so, then we need to trigger a Prune(*,G) to the old upstream neighbor.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-4.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.4 p19 (S,G) State							
	`The upstream (S,G) Join/Prune timer is used send out to override Prune(S,G) messages from peers on an upstream LAN interface`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-4.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.4 p19 (S,G) State							
	`The last RPF neighbor towards the S is stored because if the MRIB changes then the RPF neighbor towards the S may change. If it does so, then we need to trigger a new Join (S,G) to the new upstream neighbor`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-4.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.4 p19 (S,G) State							
	The last RPF neighbor towards the S is stored because if the MRIB changes then the RPF neighbor towards the S may change. If it does so, then we need to trigger a Prune(S,G) to the old upstream neighbor.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-4.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.4 p19 (S,G) State							
	If the router detects through a changed GenID in a Hello message that the upstream neighbor towards S has rebooted, then it should re-instantiate state by sending a Join(S,G).							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-4.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.1.4 p19 (S,G) State							
	Amongst other things, this is necessary for the so-called "turnaround rules" - when the RP uses (S,G) joins to stop encapsulation, and then (S,G) prunes to prevent traffic from unnecessarily reaching the RP.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-4.6 **MUST**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.1.4 p19 (S,G) State							
	The SPTbit is used to indicate whether forwarding is taking place on the (S,G) Shortest Path Tree (SPT) or on the (*,G) tree. When SPTbit is FALSE, only (*,G) forwarding state is used to forward packets from S to G.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-4.7	draft-ietf-pim-sm-v2-new-07.ps s4.1.4 p19 (S,G) State							
MUST	The SPTbit is used to indicate whether forwarding is taking place on the (S,G) Shortest Path Tree (SPT) or on the (*,G) tree. When SPTbit is TRUE, both (*,G) and (S,G) forwarding state are used.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-5.1	draft-ietf-pim-sm-v2-new-12.txt s4.2 p26 Data Packet Forwarding Rules							

MUST	if(iif == RPF_interface(S) AND UpstreamJPState(S,G) == Joined) {
	   oiflist = inherited_olist(S,G)
	   if(oiflist != NULL) {
	       restart KeepaliveTimer(S,G)
	   }
	}
	oiflist = oiflist (-) iif
	forward packet on all interfaces in oiflist
	If the SPT-bit of an (S,G) entry is set, and if incoming interface is
	the same as a matching (S,G) ifaceIn, the packet is forwarded to the
	oif-list of (S,G)							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-5.2	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.2 p26 Data Packet Forwarding Rules							

MUST	if(iif == RPF_interface(S) AND UpstreamJPState(S,G) == Joined) {
	   oiflist = inherited_olist(S,G)
	   if(oiflist != NULL) {
	       restart KeepaliveTimer(S,G)
	   }
	}
	oiflist = oiflist (-) iif
	forward packet on all interfaces in oiflist
	If the SPT-bit of an (S,G) entry is set, and if incoming interface is
	the same as a matching (S,G) ifaceIn, the packet is forwarded to the
	oif-list of (S,G)							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x

PIM-SMV6-5.3

MUST

draft-ietf-pim-sm-v2-new-12.txt s4.2 p26 Data Packet Forwarding Rules

```
 if( iif == RPF_interface(S) AND UpstreamJPState(S,G) == Joined ) {
   ...
 } else if( iif == RPF_interface(RP) AND SPTbit(S,G) == FALSE) {
   oiflist = inherited_olist(S,G,rpt)
   CheckSwitchToSpt(S,G)
 }
 oiflist = oiflist (-) iif
 forward packet on all interfaces in oiflist
On receiving multicast data packet if SPT-bit of an (S,G) entry is
cleared, and ifaceIn differs than a matching (S,G) ifaceIn but matches
with a (*,G) ifaceIn, packet is forwarded to the oif-list of (*,G)
```

Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Free BSD 10.3 untested							
Ubuntu 18.04: pass							
Free BSD 12.0 untested							

PIM-SMV6-5.4

MUST

draft-ietf-pim-sm-v2-new-12.txt s4.2 p26 Data Packet Forwarding Rules

```
 oiflist = NULL
 if( iif == RPF_interface(S) AND UpstreamJPState(S,G) == Joined ) {
    ...
 } else if( iif == RPF_interface(RP) AND SPTbit(S,G) == FALSE) {
    ...
 } else {
    # Note: RPF check failed
    if ( SPTbit(S,G) == TRUE AND iif is in inherited_olist(S,G) ) {
       send Assert(S,G) on iif
    } else if ( SPTbit(S,G) == FALSE AND
                iif is in inherited_olist(S,G,rpt) {
       send Assert(*,G) on iif
    }
 }
 oiflist = oiflist (-) iif
 forward packet on all interfaces in oiflist
On receiving multicast data packet, if incoming interface does not
match (S,G) ifaceIn or (*,G) ifaceIn, the packet is not forwarded
```

Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Free BSD 10.3 untested							
Ubuntu 18.04: pass							
Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-5.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.2 p26 Data Packet Forwarding Rules							

	`if ( SPTbit(S,G) == TRUE AND iif is in inherited_olist(S,G) ) {`
	`  send Assert(S,G) on iif`
	`} else if ( SPTbit(S,G) == FALSE AND`
	`            iif is in inherited_olist(S,G,rpt) {`
	`   send Assert(*,G) on iif`
	`}`
	`On receipt a data from S to G on interface iif, if SPT-bit is TRUE,`
	`it will send an Assert(S,G) on iif.`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-5.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.2 p26 Data Packet Forwarding Rules							

	`if ( SPTbit(S,G) == TRUE AND iif is in inherited_olist(S,G) ) {`
	`  send Assert(S,G) on iif`
	`} else if ( SPTbit(S,G) == FALSE AND`
	`            iif is in inherited_olist(S,G,rpt) {`
	`   send Assert(*,G) on iif`
	`}`
	`On receipt a data from S to G on interface iif, if SPT-bit is FALSE,`
	`it will send an Assert(*,G) on iif.`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-6.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt 4.2.2 p29 Setting and Clearing the (S,G) SPT bit							

	`Thus, when a packet arrives, the (S,G) SPTbit is updated as follows:`
	`     void`
	`     Update_SPTbit(S,G,iif) {`
	`       if ( iif == RPF_interface(S)`
	`             AND JoinDesired(S,G) == TRUE`
	`             AND ( DirectlyConnected(S) == TRUE`
	`                   OR RPF_interface(S) != RPF_interface(RP)`
	`                   OR inherited_olist(S,G,rpt) == NULL`
	`                   OR RPF'(S,G) == RPF'(*,G) ) ) {`
	`          Set SPTbit(S,G) to TRUE`
	`       }`
	`     }`
	`Here The RPF interface to S is different from the RPF interface to the`
	`RP`
	`Here RP Tree is built by (*,G) Join message`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-7.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p29 Sending Hello Messages							
	`PIM-Hello messages are sent periodically on each PIM-enabled interface. Hello messages must be sent every &lt;Hello-Period&gt; seconds.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-7.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p29 Sending Hello Messages							
	`Hello messages MUST be sent on all active interfaces, including physical point-to-point links, and are multicast to address 224.0.0.13 (the ALL-PIM-ROUTERS group).`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-7.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p29 Sending Hello Messages							
	`When PIM is enabled on an interface or a router first starts,the hello timer of that interface is set to a random value between 0 and Triggered_Hello_Delay.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-7.5 **MAY**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p30 Sending Hello Messages							
	`The neighbors will not accept Join/Prune from a router unless they have first heard a Hello message from that router. (This test is for (*,G) join state)`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-7.6 **MAY**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p30 Sending Hello Messages							
	`The neighbors will not accept Join/Prune from a router unless they have first heard a Hello message from that router. (This test is for (S,G) join state)`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-7.7 **SHOULD**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p30 Sending Hello Messages							
	`The DR_Election_Priority Option SHOULD be included in every Hello message.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-7.8 **SHOULD**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p30 Sending Hello Messages							
	`The DR_Election_Priority Option SHOULD be included in every Hello message, even if no DR election priority is explicitly configured on that interface. The default priority is 1.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-7.9 **SHOULD**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p30 Sending Hello Messages							
	`The Generation Identifier (GenID) Option SHOULD be included in all Hello messages`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-7.10 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p30 Sending Hello Messages							
	The GenID option contains a randomly generated 32-bit value that is regenerated each time PIM forwarding is started or restarted on the interface, including when the router itself restarts.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-7.11 **SHOULD**	draft-ietf-pim-sm-v2-new-12.txt s4.3.1 p30 Sending Hello Messages							
	The LAN_Prune_Delay Option SHOULD be included in all Hello messages sent on multi-access LANs.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-8.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.2 p32 DR Election							

	The function used for comparing DR "metrics" on interface I is:
	Bool dr_is_better(a,b,I) {
	  if(there is a neighbor n on I for which n.dr_priority_present
	      is false) {
	      return a.ip_address &gt; b.ip_address
	    } else {
	      return (a.dr_priority &gt; b.dr_priority) OR
	             (a.dr_priority == b.dr_priority AND
	               a.ip_address &gt; b.ip_address)
	    }
	}
	If no DR-priority option is specified in a Hello message, the neighbor with the highest IP address is elected as the DR.							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-8.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.2 p32 DR Election							

	`The function used for comparing DR "metrics" on interface I is:`
	`Bool dr_is_better(a,b,I) {`
	`  if( there is a neighbor n on I for which n.dr_priority_present`
	`      is false ) {`
	`      return a.ip_address &gt; b.ip_address`
	`    } else {`
	`      return ( a.dr_priority &gt; b.dr_priority ) OR`
	`             ( a.dr_priority == b.dr_priority AND`
	`               a.ip_address &gt; b.ip_address )`
	`    }`
	`}`
	`If DR-priority option is specified in a Hello message. The DR`
	`election priority is a 32-bit unsigned number and the numerically`
	`larger priority is always preferred.`
	`(When DUT is elected as DR)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-8.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.2 p32 DR Election							

	`The function used for comparing DR "metrics" on interface I is:`
	`Bool dr_is_better(a,b,I) {`
	`  if( there is a neighbor n on I for which n.dr_priority_present`
	`      is false ) {`
	`      return a.ip_address &gt; b.ip_address`
	`    } else {`
	`      return ( a.dr_priority &gt; b.dr_priority ) OR`
	`             ( a.dr_priority == b.dr_priority AND`
	`               a.ip_address &gt; b.ip_address )`
	`    }`
	`}`
	`If DR-priority option is specified in a Hello message. The DR`
	`election priority is a 32-bit unsigned number and the numerically`
	`larger priority is always preferred.`
	`(When ANVL is elected as DR)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-8.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.2 p32 DR Election							

	The function used for comparing DR "metrics" on interface I is:
	Bool dr_is_better(a,b,I) {
	  if(there is a neighbor n on I for which n.dr_priority_present
	      is false) {
	      return a.ip_address &gt; b.ip_address
	    } else {
	      return (a.dr_priority &gt; b.dr_priority) OR
	             (a.dr_priority == b.dr_priority AND
	               a.ip_address &gt; b.ip_address)
	    }
	}
	If DR-priority option is specified in a Hello message, the neighbor
	with the DR-priority is equal to that of the others then the highest IP
	address is elected as the DR.
	(When DUT is elected as DR)							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-8.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.2 p32 DR Election							

	The function used for comparing DR "metrics" on interface I is:
	Bool dr_is_better(a,b,I) {
	  if(there is a neighbor n on I for which n.dr_priority_present
	      is false) {
	      return a.ip_address &gt; b.ip_address
	    } else {
	      return (a.dr_priority &gt; b.dr_priority) OR
	             (a.dr_priority == b.dr_priority AND
	               a.ip_address &gt; b.ip_address)
	    }
	}
	If DR-priority option is specified in a Hello message, the neighbor
	with the DR-priority is equal to that of the others then the highest IP
	address is elected as the DR.
	(When ANVL is elected as DR)							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-8.6 **MAY**	draft-ietf-pim-sm-v2-new-12.txt s4.3.2 p32 DR Election							
	A router's idea of the current DR on an interface can change when a neighbor times out.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-8.7 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.3.2 p32 DR Election							
	`A router's idea of the current DR on an interface can change when a router's own DR priority changes.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-8.8 **MUST**	draft-ietf-pim-v2-new-07.txt s4.3.2 p32 DR Election							
	`The Neighbor Liveness Timer (NLT(N,I)) is reset to Hello_Holdtime (from the Hello Holdtime option) whenever a Hello message is received containing a Holdtime option.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-8.9 **MAY**	draft-ietf-pim-smi-v2-new-07.txt s4.3.2 p32 DR Election							

	`A router's idea of the current DR on an interface can change when a PIM-Hello message is received, when a neighbor times out, or when a router's own DR priority changes. If the router becomes the DR or ceases to be the DR, this will normally cause the DR Register state-machine to change state.`
	`(Here selection of the new DR to be one with the highest IP address)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-10.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.4 p35 PIM Register Messages							
	`The Designated Router (DR) on a LAN or point-to-point link encapsulates multicast packets from local sources to the RP for the relevant group unless it recently received a Register Stop message for that (S,G) or (*,G) from the RP.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-10.2	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.4 p35 PIM Register Messages							
MUST	The Designated Router (DR) on a LAN or point-to-point link encapsulates multicast packets from local sources to the RP for the relevant group unless it recently received a Register Stop message for that (S,G) or (*,G) from the RP.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-10.3	draft-ietf-pim-sm-v2-new-12.txt s4.4 p35 PIM Register Messages							
MUST	The Designated Router (DR) on a LAN or point-to-point link does not encapsulates multicast packets from local sources to the RP for the relevant group if it recently received a Register Stop message for that (S,G) from the RP.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-10.4	draft-ietf-pim-sm-v2-new-12.txt s4.4 p35 PIM Register Messages							
MUST	When the DR receives a Register Stop message from the RP, it starts a Register Stop timer to maintain this state. Just before the Register Stop timer expires, the DR sends a Null-Register Message to the RP to allow the RP to refresh the Register Stop information at the DR.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.1	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							
MUST	In Join(J) state if DR receives RegisterStop Message, then it will go to Prune(P) state by removing register tunnel and set Register-Stop Timer.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-11.2	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							
MUST	`In Join(J) state if CouldRegister(S,G) becomes false then it will go to NoInfo(NI) State Here CouldRegister(S,G) -&gt; FALSE is achieved by making I_am_DR(RPF_interface(S))-&gt;FALSE`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.3	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							
MUST	`In Join(J) state if RP(G) changes, then the DR updates Register tunnel`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.4	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							
MUST	`In Join Pending(JP) state if RegStop timer expires then the DR will go to Join(J) state by adding the register tunnel`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.5	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							
MUST	`In Join Pending(JP) state if RP changed then the DR will go to Join(J) state by adding the register tunnel and cancel the Register-Stop Timer.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-11.6	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							

MUST	`In Join Pending(JP) state if CouldRegister(S,G) becomes false then it will go to NoInfo(NI) State`
	`Here CouldRegister(S,G) -&gt; FALSE is achieved by making I_am_DR(RPF_interface(S))-&gt;FALSE`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.7	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							
MUST	`In Join Pending(JP) state if RegStop is received The the DR goes to Prune(P) state and set RegStop timer to randomised RSI - probetime`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.8	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							

MUST	`In Prune(P) state if CouldRegister(S,G) becomes false then it will go to NoInfo(NI) State`
	`Here CouldRegister(S,G) -&gt; FALSE is achieved by making I_am_DR(RPF_interface(S))-&gt;FALSE`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.9	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							
MUST	`In Prune(P) state if RP(G) changes, then the DR goes to Join(J) state and adds Register Channel`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-11.10 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p37 Sending Register Messages from the DR							

	`In NoInfo(NI) if CouldRegister(S,G) becomes true then DR will go to Join(J) State`
	`Here CouldRegister(S,G) -&gt; TRUE is achieved by making I_am_DR(RPF_interface(S))-&gt;TRUE`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-11.11 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.4.1 p39 Sending Register Messages from the DR							
	`A RegisterStop(*,G) should be treated as a RegisterStop(S,G) for all existing (S,G) Register state machines`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-12.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p40 Receiving Register Messages at the RP							

	`When an RP receives a Register message, the course of action is decided according to the following pseudocode:`
	`Packet_arrives_on_rp_tunnel( pkt ) {`
	`    ...`
	`    if(( inherited_olist(S,G) == NULL ) OR SPTbit(S,G)) {`
	`       send RegisterStop(S,G) to outer.src`
	`    } else {`
	`        ...`
	`      }`
	`    }`
	`    ...`
	`}`
	`If ( inherited_olist(S,G) == NULL )then RP send RegisterStop(S,G) to outer.src i.e., the DRs address.`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-12.2 **MUST**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p40 Receiving Register Messages at the RP							

```
When an RP receives a Register message, the course of action is
decided according to the following pseudocode:
Packet_arrives_on_rp_tunnel( pkt ) {
   ...
    if(( inherited_olist(S,G) == NULL ) OR SPTbit(S,G)) {
       send RegisterStop(S,G) to outer.src
    } else {
      if( ! pkt.NullRegisterBit ) {
        decapsulate and pass the inner packet to the normal
        forwarding path for forwarding on the (*,G) tree.
      }
    }
   ...
}
If (S,G) entry with cleared (0) SPT bit exists, and received Register
without Null-Register-Bit set to 1 then RP decapsulate and pass the
inner packet to the normal forwarding path.
```

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-12.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p40 Receiving Register Messages at the RP							

```
When an RP receives a Register message, the course of action is
decided according to the following pseudocode:
Packet_arrives_on_rp_tunnel( pkt ) {
   ...
    if(( inherited_olist(S,G) == NULL ) OR SPTbit(S,G)) {
       send RegisterStop(S,G) to outer.src
    } else {
      if( ! pkt.NullRegisterBit ) {
        decapsulate and pass the inner packet to the normal
        forwarding path for forwarding on the (*,G) tree.
      }
    }
   ...
}
If (inherited_olist(S,G) != NULL) and there is no (S,G) entry
and received Register has Null-Register-Bit set to 0 then
RP decapsulate and pass the inner packet to the normal
forwarding path for forwarding on the (*,G) tree.
```

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

PIM-SMV6-12.4 draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p40 Receiving Register Messages at the RP

MUST

```
When an RP receives a Register message, the course of action is
decided according to the following pseudocode:
Packet_arrives_on_rp_tunnel( pkt ) {
    ...
    if( I_am_RP( G ) && outer.dst == RP(G) ) {
    ...
    } else {
        send RegisterStop(S,G) to outer.src
        # Note (*)
    }
}
Here it is tested if (I_am_RP( G ) -&gt; FALSE) RP sent a Register Stop
Message
```

Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Free BSD 10.3 untested							
Ubuntu 18.04: inconclusive							
Free BSD 12.0 untested							

PIM-SMV6-12.5 draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p40 Receiving Register Messages at the RP

MUST

```
When an RP receives a Register message, the course of action is
decided according to the following pseudocode:
Packet_arrives_on_rp_tunnel( pkt ) {
    ...
    if( I_am_RP( G ) && outer.dst == RP(G) ) {
    ...
    } else {
        send RegisterStop(S,G) to outer.src
        # Note (*)
    }
}
Here it is tested if (I_am_RP(G) -&gt; FALSE) RP does not forward the data
```

Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Free BSD 10.3 untested							
Ubuntu 18.04: inconclusive							
Free BSD 12.0 untested							

PIM-SMV6-12.6

MUST

draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p40 Receiving Register Messages at the RP

```
When an RP receives a Register message, the course of action is
decided according to the following pseudocode:
Packet_arrives_on_rp_tunnel(pkt) {
      ...
    if (I_am_RP(G) && outer.dst == RP(G)) {
      ...
    } else {
      send RegisterStop(S,G) to outer.src
      # Note (*)
    }
}
Here (outer.dst == RP(G))-&gt;FALSE
```

Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Free BSD 10.3 untested							
Ubuntu 18.04: inconclusive							
Free BSD 12.0 untested							

PIM-SMV6-12.7

MUST

draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p40 Receiving Register Messages at the RP

```
When an RP receives a Register message, the course of action is
decided according to the following pseudocode:
Packet_arrives_on_rp_tunnel( pkt ) {
    ...
    if( I_am_RP( G ) && outer.dst == RP(G) ) {
    ...
      if(( inherited_olist(S,G) == NULL ) OR SPTbit(S,G)) {
          ...
        } else {
            if( ! pkt.NullRegisterBit ) {
                decapsulate and pass the inner packet to the normal
                forwarding path for forwarding on the (*,G) tree.
            }
        }
    }

}
Here pkt.NullRegisterBit -&gt; TRUE
```

Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Free BSD 10.3 untested							
Ubuntu 18.04: inconclusive							
Free BSD 12.0 untested							

PIM-SMV6-12.8

MUST

draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p41 Receiving Register Messages at the RP

```
Just like any forwarded packet, the HopLimit of the original data packet is
decremented after it is decapsulated from the Register Tunnel.
```

Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
Free BSD 10.3 untested							
Ubuntu 18.04: inconclusive							
Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-12.9	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.4.2 p41 Receiving Register Messages at the RP							
MUST	Just like any forwarded packet, the HopLimit of the original data packet is decremented after it is decapsulated from the Register Tunnel.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-14.1	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p46 Receiving (*,G) Join/Prune Messages							
MAY	If a router has no RP information (e.g. has not recently received a BSR message) then it may choose to accept Join(*,G) and treat the RP in the message as RP(G).							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-14.2	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p46 Receiving (*,G) Join/Prune Messages							
MAY	If a router has no RP information (e.g. has not recently received a BSR message) then it may choose to accept Prune(*,G) and treat the RP in the message as RP(G).							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-14.3	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p47 Receiving (*,G) Join/Prune Messages							
MUST	In NoInfo(NI) state by receiving Prune(*,G) message the (*,G) downstream state machine on interface I remains in the NoInfo state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-14.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p47-48 Receiving (*,G) Join/Prune Messages							
	In NoInfo(NI) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I transitions to the Join state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.5 **MUST**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
	In NoInfo(NI) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I transitions to the Join state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
	In Join(J) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I remains in Join state, and the Expiry Timer (ET) is restarted, set to maximum of its current value and the HoldTime from the triggering Join/Prune message. (When current value is smaller than HoldTime from the triggering Join/Prune message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.7 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
	In Join(J) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I remains in Join state, and the Expiry Timer (ET) is restarted, set to maximum of its current value and the HoldTime from the triggering Join/Prune message. (When current value is greater than HoldTime from the triggering Join/Prune message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-14.8	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
MUST	In Join(J) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I remains in Join state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.9	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
MUST	In Join(J) state by receiving Prune(*,G) message the (*,G) downstream state machine on interface I transitions to the PrunePending state. The PrunePending timer is started; if the router has one neighbor on that interface; then it is set to zero causing it to expire immediately.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.10	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
MUST	In Join(J) state by receiving Prune(*,G) message the (*,G) downstream state machine on interface I transitions to the PrunePending state. The PrunePending timer is started; it is set to the J/P_Override_Interval(I) if the router has more than one neighbor on that interface;							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.11	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
MUST	In Join(J) state if the Expiry Timer for the (*,G) downstream state machine on interface I expires. The (*,G) downstream state machine on interface I transitions to the NoInfo state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-14.12 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p47 Receiving (*,G) Join/Prune Messages							
	`In PrunePending(PP) state by receiving Prune(*,G) message the (*,G) downstream state machine on interface I remains into the PrunePending state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.13 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
	`In PrunePending(PP) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I transitions to the Join state. The PrunePending timer is canceled (without triggering an expiry event).`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.14 **MUST**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
	`In PrunePending(PP) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I transitions to the Join state. The PrunePending timer is canceled (without triggering an expiry event).`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.15 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
	`In PrunePending(PP) state by receiving Join(*,G) message the (*,G) downstream state machine on interface I transitions to the Join state. The Expiry Timer is restarted, set to maximum of its current value and the HoldTime from the triggering Join/Prune message.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-14.16 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p49 Receiving (*,G) Join/Prune Messages							
	In PrunePending(PP) state if the Expiry Timer for the (*,G) downstream state machine on interface I expires. The (*,G) downstream state machine on interface I transitions to the NoInfo state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-14.17 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p49 Receiving (*,G) Join/Prune Messages							
	In PrunePending(PP) state if the PrunePending Timer for the (*,G) downstream state machine on interface I expires. The (*,G) downstream state machine on interface I transitions to the NoInfo state. A PruneEcho(*,G) is sent onto the subnet connected to interface I.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p50 Receiving (S,G) Join/Prune Messages							
	In NoInfo(NI) state by receiving Prune(S,G) message the (S,G) downstream state machine on interface I remains in the NoInfo state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
	In NoInfo(NI) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I transitions to the Join state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-15.3 **MUST**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
	In NoInfo(NI) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I transitions to the Join state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
	In Join(J) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I remains in Join state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
	In Join(J) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I remains in Join state, and the Expiry Timer (ET) is restarted, set to maximum of its current value and the HoldTime from the triggering Join/Prune message. (When current value is greater than HoldTime from the triggering Join/Prune message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
	In Join(J) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I remains in Join state, and the Expiry Timer (ET) is restarted, set to maximum of its current value and the HoldTime from the triggering Join/Prune message. (When current value is smaller than HoldTime from the triggering Join/Prune message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-15.7	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
MUST	`In Join(J) state by receiving Prune(S,G) message the (S,G) downstream state machine on interface I transitions to the PrunePending state. The PrunePending timer is started; if the router has one neighbor on that interface; then it is set to zero causing it to expire immediately.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.8	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
MUST	`In Join(J) state by receiving Prune(S,G) message the (S,G) downstream state machine on interface I transitions to the PrunePending state. The PrunePending timer is started; it is set to the J/P_Override_Interval(I) if the router has more than one neighbor on that interface;`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.9	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p51 Receiving (S,G) Join/Prune Messages							
MUST	`In Join(J) state if the Expiry Timer for the (S,G) downstream state machine on interface I expires. The (S,G) downstream state machine on interface I transitions to the NoInfo state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.10	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p50 Receiving (S,G) Join/Prune Messages							
MUST	`In PrunePending(PP) state by receiving Prune(S,G) message the (S,G) downstream state machine on interface I remains into the PrunePending state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-15.11	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p52 Receiving (S,G) Join/Prune Messages							
MUST	`In PrunePending(PP) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I transitions to the Join state. The PrunePending timer is canceled (without triggering an expiry event).`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.12	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p52 Receiving (S,G) Join/Prune Messages							
MUST	`In PrunePending(PP) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I transitions to the Join state. The PrunePending timer is canceled (without triggering an expiry event).`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.13	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p52 Receiving (S,G) Join/Prune Messages							
MUST	`In PrunePending(PP) state by receiving Join(S,G) message the (S,G) downstream state machine on interface I transitions to the Join state. The Expiry Timer is restarted, set to maximum of its current value and the HoldTime from the triggering Join/Prune message.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-15.14	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p52 Receiving (S,G) Join/Prune Messages							
MUST	`In PrunePending(PP) state if the Expiry Timer for the (S,G) downstream state machine on interface I expires. The (S,G) downstream state machine on interface I transitions to the NoInfo state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-15.15 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.3 p52 Receiving (S,G) Join/Prune Messages							
	`In PrunePending(PP) state if the PrunePending Timer for the (S,G) downstream state machine on interface I expires. The (S,G) downstream state machine on interface I transitions to the NoInfo state. A PruneEcho(S,G) is sent onto the subnet connected to interface I.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p54 Receiving (S,G,rpt) Join/Prune Messages draft-ietf-pim-sm-v2-new-07.ps s4.5.4 p40 Figure 5: Downstream per-interface (S,G,rpt) state-machine							
	`In NoInfo(NI) state by receiving Join(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I remains in the NoInfo state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p55 Receiving (S,G,rpt) Join/Prune Messages							
	`In NoInfo(NI) state by receiving Prune(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I transitions to PrunePending(PP) state. The PrunePending timer is started; it is set to the J/P_Override_Interval(I) if the router has more than one neighbor on that interface; otherwise it is set to causing it to expire immediately (Here DUT has only one downstream neighbor)`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p55 Receiving (S,G,rpt) Join/Prune Messages							
	`In NoInfo(NI) state by receiving Prune(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I transitions to PrunePending(PP) state. The PrunePending timer is started; it is set to the J/P_Override_Interval(I) if the router has more than one neighbor on that interface`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-16.4	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p54 Receiving (S,G,rpt) Join/Prune Messages							
MUST	In PrunePending(PP) state by receiving Prune(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I remains in the PrunePending(PP) state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.5	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p55 Receiving (S,G,rpt) Join/Prune Messages							
MUST	In PrunePending (PP) state by receiving Join(*,G) message the (S,G,rpt) downstream state machine on interface I transitions to the PrunePendingTmp(PP') state. If the (*,G) message does not contain (S,G,rpt) Prune information the downstream state machine on interface I transitions to NoInfo state							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.6	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p55 Receiving (S,G,rpt) Join/Prune Messages							
MUST	In PrunePending (PP) state by receiving Join(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I transitions to NoInfo state. ET and PPT are canceled.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.7	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p55-56 Receiving (S,G,rpt) Join/Prune Messages							
MUST	In PrunePending (PP) state if the PrunePending Timer for the (S,G,rpt) downstream state machine on interface I expires. The (S,G,rpt) downstream state machine on interface I transitions to the Pruned state							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-16.8 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p56 Receiving (S,G,rpt) Join/Prune Messages							
	In Pruned(P) state by receiving Join(*,G) message the (S,G,rpt) downstream state machine on interface I transitions to PruneTmp state. If the (*,G) message does not contain (S,G,rpt) Prune information the downstream state machine on interface I transitions to NoInfo state (Here DUT has only one downstream neighbor)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.9 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p56 Receiving (S,G,rpt) Join/Prune Messages							
	In Pruned (P) state by receiving Join(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I transitions to NoInfo state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.10 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p56 Receiving (S,G,rpt) Join/Prune Messages							
	In Pruned(P) state by receiving Prune(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I remains in Pruned state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-16.11 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p56 Receiving (S,G,rpt) Join/Prune Messages							
	In Pruned(P) state by receiving Prune(S,G,rpt) message the (S,G,rpt) downstream state machine on interface I remains in Pruned state. The Expiry Timer (ET) is restarted, set to maximum of its current value and the HoldTime from the triggering Join/Prune message.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-16.12 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.4 p56 Receiving (S,G,rpt) Join/Prune Messages							
	`In Pruned(P) state if the Expiry Timer for the (S,G,rpt) downstream state machine on interface I expires. The (S,G,rpt) downstream state machine on interface I transitions to the NoInfo state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-18.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.6 p64 Sending (*,G) Join/Prune Messages							

	`JoinDesired(*,G) becomes True`
	`The downstream state for (*,G) has changed so that at least one interface is in immediate_olist(*,G), making JoinDesired(*,G) become True. The upstream (*,G) state machine transitions to Joined state. Send Join(*,G) to the appropriate upstream neighbor, which is RPF'(*,G).`
	`(Here Join List verified)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-18.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.6 p64 Sending (*,G) Join/Prune Messages s4.10.5.1, p116 Group Set Source List Rules							

	`JoinDesired(*,G) becomes True`
	`The downstream state for (*,G) has changed so that at least one interface is in immediate_olist(*,G), making JoinDesired(*,G) become True. The upstream (*,G) state machine transitions to Joined state. Send Join(*,G) to the appropriate upstream neighbor, which is RPF'(*,G).`
	`(Here WC and RPT Bit are checked)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-18.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.6 p64 Sending (*,G) Join/Prune Messages							

	`JoinDesired(*,G) becomes False`
	`The downstream state for (*,G) has changed so no interface is in immediate_olist(*,G), making JoinDesired(*,G) become False. The upstream (*,G) state machine transitions to NotJoined state. Send Prune(*,G) to the appropriate upstream neighbor, which is RPF'(*,G).`
	`(Here Prune List verified)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-18.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.6 p64 Sending (*,G) Join/Prune Messages s4.10.5.1, p116 Group Set Source List Rules							

	`JoinDesired(*,G) becomes False`
	`The downstream state for (*,G) has changed so no interface is in immediate_olist(*,G), making JoinDesired(*,G) become False. The upstream (*,G) state machine transitions to NotJoined state. Send Prune(*,G) to the appropriate upstream neighbor, which is RPF'(*,G).`
	`(Here WC and RPT Bit are checked)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-18.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.6 p64-65 Sending (*,G) Join/Prune Messages							
	`When the upstream (*,G) state-machine is in Joined state, if the Join Timer (JT) expires, indicating time to send a Join(*,G). Send Join(*,G) to the appropriate upstream neighbor, which is RPF'(*,G). Restart the Join Timer (JT) to expire after t_periodic seconds.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-18.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.6 p66 Sending (*,G) Join/Prune Messages							
	`When the upstream (*,G) state-machine is in Joined state, if the RPF'(*,G) GenID changes then the upstream (*,G) state machine remains in Joined state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-19.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.7 p69 Sending (S,G) Join/Prune Messages							

	`The downstream state for (S,G) has changed so that at least one interface is in inherited_olist(S,G), making JoinDesired(S,G) become True.`
	`(Verify (S,G) Join List contains the Source Address in Join List)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-19.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.7 p69 Sending (S,G) Join/Prune Messages s4.10.5.1, p116 Group Set Source List Rules							
	The source address S (with cleared RPT and WC bits) is included in the join list of a periodic Join/Prune for an active (S,G) entry with cleared RPT-bit flag and oif-list is not null. (Here WC and RPT Bit are checked)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-19.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.7 p69 Sending (S,G) Join/Prune Messages							
	JoinDesired(S,G) becomes False The downstream state for (S,G) has changed so no interface is in inherited_olist(S,G), making JoinDesired(S,G) become False. The upstream (S,G) state machine transitions to NotJoined state. Send Prune(S,G) to the appropriate upstream neighbor, which is RPF'(S,G) (Here Prune List verified)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-19.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.7 p69 Sending (S,G) Join/Prune Messages s4.10.5.1, p116 Group Set Source List Rules							
	JoinDesired(S,G) becomes False The downstream state for (S,G) has changed so no interface is in inherited_olist(S,G), making JoinDesired(S,G) become False. The upstream (S,G) state machine transitions to NotJoined state. Send Prune(S,G) to the appropriate upstream neighbor, which is RPF'(S,G) (Here WC and RPT Bit are checked)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-19.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.7 p69 Sending (S,G) Join/Prune Messages							
	When the upstream (S,G) state-machine is in Joined state, if the Join Timer (JT) expires, indicating time to send a Join(S,G). Send Join(S,G) to the appropriate upstream neighbor, which is RPF'(S,G). Restart the Join Timer (JT) to expire after t_periodic seconds.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-19.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.6 p66 Sending (S,G) Join/Prune Messages							
	When the upstream (S,G) state-machine is in Joined state, if the RPF'(S,G) GenID changes then the upstream (S,G) state machine remains in Joined state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-20.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.9 p75-76 State Machine for (S,G,rpt) Triggered Messages							
	In "NotPruned" State, if PruneDesired(S,G,rpt)-&gt;TRUE the action is to send a Prune(S,G,rpt) to RPF'(S,G,rpt)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-20.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.5.9 p76 State Machine for (S,G,rpt) Triggered Messages							
	If the router is in the Pruned(S,G,rpt) state, and PruneDesired(S,G,rpt) changes to FALSE, this could be because the router no longer has RPTJoinDesired(G) true, or it now wishes to receive traffic from S again. If it is not the former the action is to send a Join(S,G,rpt) to RPF'(S,G,rpt)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p77 (S,G) Assert State Machine							
	This router has lost an (S,G) assert on interface I. It must not forward packets for G onto interface I.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-21.2 **MUST**	NEGATIVE: draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p77 (S,G) Assert State Machine							
	`This router has lost an (S,G) assert on interface I. It must not forward packets for G onto interface I.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p77 (S,G) Assert State Machine							

	`The winning router sends an Assert message containing its own metric to that outgoing interface(State machine)`
	`(This is performed with (S,G)-(S,G) assert`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.4 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p80-81 (S,G) Assert Message State Machine							
	`When in NoInfo state, if an inferior assert is received for (S,G) with the RPT bit cleared and CouldAssert(S,G,I) == TRUE, We transition to the "I am Assert Winner" state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p80-81 (S,G) Assert Message State Machine							
	`When in NoInfo state, if an assert is received for (S,G) with the RPT bit set(it's a (*,G) assert) and CouldAssert(S,G,I) == TRUE, We Send Assert(S,G).`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-21.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p80-81 (S,G) Assert Message State Machine							
	When in NoInfo state, if an (S,G) data packet comes on Interface I and CouldAssert(S,G,I) == TRUE, We transition to the "I am Assert Winner" state							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-21.7 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p81 (S,G) Assert Message State Machine							
	When in NoInfo state, if an (S,G) data packet comes on Interface I and CouldAssert(S,G,I) == TRUE, we Send Assert(S,G)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.8 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p81 (S,G) Assert Message State Machine							
	When in "I am Assert Winner" State, if we receive an (S,G) assert that has a worse metric than our own. Whoever sent the assert is in error, and so we remains in "I am Assert Winner" State							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.9 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p81 (S,G) Assert Message State Machine							
	When in "I am Assert Winner" State, if we receive an (S,G) assert that has a worse metric than our own. Whoever sent the assert is in error, and so we re-send an (S,G) Assert							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-21.10 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p81 (S,G) Assert Message State Machine							
	When in "I am Assert Winner" State, if we receive an (S,G) assert that has a worse metric than our own. Whoever sent the assert is in error, and so we re-send an (S,G) Assert and so we set the timer to &lt;Assert_Time&gt; - &lt;Assert_Override_Interval&gt;							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.11 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p81 (S,G) Assert Message State Machine							
	When in "I am Assert Winner" State, if we receive an (*,G) assert mentioning S that has a worse metric than our own. Whoever sent the assert is in error, and so we remains in "I am Assert Winner" State							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.12 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p81 (S,G) Assert Message State Machine							
	When in "I am Assert Winner" State, if we receive an (*,G) assert mentioning S that has a worse metric than our own. Whoever sent the assert is in error, and so we re-send an (S,G) Assert							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-21.13 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p81 (S,G) Assert Message State Machine							
	When in "I am Assert Winner" State, if we receive an (*,G) assert mentioning S that has a worse metric than our own. Whoever sent the assert is in error, and so we set the timer to &lt;Assert_Time&gt; - &lt;Assert_Override_Interval&gt;							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-21.14 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p82 (S,G) Assert Message State Machine							
	`When in "I am Assert Winner" State, if we receive an (S,G) assert that has a better metric than our own, we transition to "I am Assert Loser" state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.15 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p82 (S,G) Assert Message State Machine							
	`When in "I am Assert Winner" State, if CouldAssert(S,G,I) become false, we send a "canceling assert" with an infinite metric.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.16 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p82 (S,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, we receive an assert that is better than that of the current assert winner.  We stay in Loser state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.17 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p82 (S,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, we receive an assert from the current assert winner that is better than our own metric for this (S,G) (although the metric may be worse than the winner's previous metric). We stay in Loser state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-21.18 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p82 (S,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, if we receive an assert from the current assert winner that is worse than our own metric for this group, we transition to NoInfo state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.19 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p82 (S,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, the (S,G) assert timer expires, we transition to NoInfo state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.20 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p82-83 (S,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, we receive a Hello message from the current winner reporting a different GenID from the one it previously reported, we transition to the "NoInfo" state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.21 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p83 (S,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, my_assert_metric(S,G,I) has changed so that now my assert metric for (S,G) is better than the metric we have stored for current assert winner. We transition to NoInfo state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-21.22	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p83 (S,G) Assert Message State Machine							
MUST	When in "I am Assert Loser" State, interface I used to be the RPF interface for S, and now it is not. We transition to NoInfo state, deleting this (S,G) assert state action as delete assert info							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-21.23	draft-ietf-pim-sm-v2-new-12.txt s4.6.1 p77 (S,G) Assert State Machine							
MUST	When in "I am Assert Loser" State, we receive a Join(S,G) that has the Upstream Neighbor Address field set to one my IP address on interface I. The action is to transition to NoInfo state							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.1	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p84 (*,G) Assert State Machine							
MUST	This router has lost an (*,G) assert on interface I. It must not forward packets for G onto interface I.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.2	NEGATIVE: draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p84 (*,G) Assert State Machine							
MUST	This router has lost an (*,G) assert on interface I. It must not forward packets for G onto interface I.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-22.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p86 (*,G) Assert State Machine							

	`The winning router sends an Assert message containing its own metric`
	`to that outgoing interface(State machine)`
	`(this is performed with (*,G)-(*,G) assert`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.4 **MUST**	draft-ietf-pim-v2-sm-01.txt s4.6.2 p88 (*,G) Assert Message State Machine, s4.10.6 p121 Assert Message Format							

	`Receive Inferior Assert`
	`We receive a (*,G) assert that has a worse metric than our own.`
	`Whoever sent the assert has lost, and so we re-send a (*,G) Assert,`
	`and restart the timer.`
	`(Here check that RPT bit is set for the Assert sent by Assert Winner)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p88 (*,G) Assert Message State Machine							

	`When in NoInfo state, if an (S,G) data packet comes on Interafce I and`
	`CouldAssert(*,G,I) == TRUE, we transition to the "I am Assert Winner"`
	`state`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-22.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p88 (*,G) Assert Message State Machine							

	`When in NoInfo state, if an (S,G) data packet comes on Interface I and`
	`CouldAssert(*,G,I) == TRUE, we Send Assert(*,G)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-22.7 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p88 (*,G) Assert Message State Machine							
	`When in "I am Assert Winner" State, we receive a (*,G) assert that has a better metric than our own. We transition to "I am Assert Loser" state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.8 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p89 (*,G) Assert Message State Machine							
	`When in "I am Assert Winner" State,if CouldAssert(*,G,I) become false, we send a "canceling assert" with an infinite metric.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.9 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p89 (*,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, we receive a (*,G) assert that is better than that of the current assert winner. We stay in Loser state.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.10 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p89 (*,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, we receive a (*,G) assert from the current assert winner that is better than our own metric for this group (although the metric may be worse than the winner's previous metric). We stay in Loser state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-22.11 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p89 (*,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, if we receive an assert from the current assert winner that is worse than our own metric for this group we transition to NoInfo state`							
	Free BSD 10.3 untested							
	🟥 Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.12 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p89 (*,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, the (*,G) assert timer expires, we transition to NoInfo state`							
	Free BSD 10.3 untested							
	🟥 Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.13 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p89 (*,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, we receive a Hello message from the current winner reporting a different GenID from the one it previously reported, we transition to the "NoInfo" state`							
	Free BSD 10.3 untested							
	🟥 Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.14 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p90 (*,G) Assert Message State Machine							
	`When in "I am Assert Loser" State, My route metric rpt_assert_metric(G,I) has changed so that now my assert metric for (*,G) is better than the metric we have stored for current assert winner. We transition to NoInfo state`							
	Free BSD 10.3 untested							
	🟥 Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-22.15	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p96 (*,G) Assert Message State Machine							
MUST	`When in "I am Assert Loser" State, interface I used to be the RPF interface for RP, and now it is not. We transition to NoInfo state, deleting this (*,G) assert state action as delete assert info`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-22.16	draft-ietf-pim-sm-v2-new-12.txt s4.6.2 p90 (*,G) Assert Message State Machine							
MUST	`When in "I am Assert Loser" State, we receive a Join(*,G) that has the Upstream Neighbor Address field set to one my IP address on interface I. The action is to transition to NoInfo state`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-23.1	draft-ietf-pim-sm-v2-new-12.txt s4.6.3 p91 Assert Metrics							

MUST	`If all fields are equal, the IP address of the router that sourced the Assert message is used as a tie-breaker, with the highest IP address winning.`
	`(This is for (*,G) Assert)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-23.2	draft-ietf-pim-sm-v2-new-12.txt s4.6.3 p91 Assert Metrics (This is for (S,G) Assert)							
MUST	`If all fields are equal, the IP address of the router that sourced the Assert message is used as a tie-breaker, with the highest IP address winning.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-24.2	draft-ietf-pim-sm-v2-new-12.txt s4.8.1 p100 Group-to-RP Mapping							
MAY	`if the set of possible group-range-to-RP mappings changes, each router will need to check whether any existing groups are affected. This may, for example, cause a DR or acting DR to re-join a group to the new RP. (This is done for (*,G) Join)`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-25.2	draft-ietf-pim-sm-v2-new-12.txt s4.9 p102 Source-Specific Multicast							
MUST	`A range of multicast addresses, currently 232.0.0.0/8 in IPv4, is reserved for SSM, and the choice of semantics is determined by the multicast group address in both data packets and PIM messages. ((S,G) Join Message with group address is in SSM range)`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-28.1	draft-ietf-pim-sm-v2-new-12.txt s4.10 p104 PIM Packet Formats							
MUST	`All PIM control messages have IP protocol number 103.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-28.2	draft-ietf-pim-sm-v2-new-12.txt s4.10 p104 PIM Packet Formats							
MUST	`Reserved field is set to 0 on transmission`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-28.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10 p105 PIM Packet Formats							
	The checksum is a standard IP checksum,i.e. the 16-bit one's Complement of the one's complement sum of the entire PIM message, excluding the "Multicast data packet" section of the Register message.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-29.1 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p106 Encoded Source and Group Address Formats							
	If the message is sent for a single group then the Mask length must equal the address length in bits for the given Address Family and Encoding Type.(e.g.128 for native IPv6 encoding)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-29.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p106 Encoded Source and Group Address Formats							
	[B]idirectional PIM indicates the group range should use Bidirectional PIM. For PIM-SM defined in this specification, this bit MUST be zero.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-29.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p106 Encoded Source and Group Address Formats							
	Admin Scope [Z]one indicates the group range is an admin scope zone. This is used in the Bootstrap Router Mechanism only. For all other purposes, this bit is set to zero (Here we are considering Non-BSR message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-29.4 **MUST**	NEGATIVE draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p106 Encoded Source and Group Address Formats							
	`Admin Scope [Z]one indicates the group range is an admin scope zone. This is used in the Bootstrap Router Mechanism only. For all other purposes, this bit is set to zero and ignore on receipt`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-29.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p107 Encoded Source and Group Address Formats							
	`The Sparse bit is a 1 bit value, set to 1 for PIM-SM.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-29.6 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p107 Encoded Source and Group Address Formats							
	`The WC(or WildCard) bit is a 1 bit value for use with PIM Join/Prune messages. (S,G) source list entries have the Source-Address set to the address of the source S, the Source-Address Mask-Len set to the full length of the IP address and have both the WC and RPT bits of the Encoded-Source-Address cleared.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-29.7 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p107 Encoded Source and Group Address Formats							
	`The RPT (or Rendezvous Point Tree) bit is a 1 bit value for use with PIM Join/Prune messages (see section 4.10.5.1 ). If the WC bit is 1, the RPT bit MUST be 1.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-29.8	draft-ietf-pim-sm-v2-new-12.txt s4.10.1 p107 Encoded Source and Group Address Formats							

MUST	`The RPT (or Rendezvous Point Tree) bit is a 1 bit value for use with`
	`PIM Join/Prune messages (see section 4.10.5.1 ). If the WC bit is 1,`
	`the RPT bit MUST be 1.`							

	Free BSD 10.3 untested							
	🟩 Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-30.1	draft-ietf-pim-sm-v2-new-12.txt s4.10.2 p109 Hello Message Format							

SHOULD	`Hello messages with a Holdtime value set to '0' are also sent by`
	`a router on an interface about to go down... These are effectively`
	`goodbye messages and the receiving routers should immediately time`
	`out the neighbor information for the sender.`
	`(Here the testing is done on whether DUT correctly`
	`times out a neighbor)`							

	Free BSD 10.3 untested							
	🟩 Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-30.2	draft-ietf-pim-sm-v2-new-12.txt s4.10.2 p109 Hello Message Format							

MUST	`Hello messages with a Holdtime value set to `0' are also sent by`
	`a router on an interface changing IP address`							

	Free BSD 10.3 untested							
	🟥 Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-30.3	draft-ietf-pim-sm-v2-new-12.txt s4.10.2 p109 Hello Message Format							

MUST	`Hello messages with a Holdtime value set to `0' are also sent by`
	`a router on an interface about to go down`							

	Free BSD 10.3 untested							
	🟥 Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-31.1	draft-ietf-pim-sm-v2-new-12.txt s4.10.3 p111 Register Message Format							
MUST	`The checksum for Registers is done only on first 8 bytes of packet, including the PIM header and the next 4 bytes, excluding the data packet portion`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-31.2	draft-ietf-pim-sm-v2-new-12.txt s4.10.3 p111 Register Message Format							
MUST	`If the router is a DR for a source that it is directly connected to, it sets the B bit to 0 in the Register message`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-32.1	draft-ietf-pim-sm-v2-new-12.txt s4.10.4 p112 RegisterStop Message							
MUST	`For Register-Stops, the Mask Len field contains full address length * 8 (e.g. 128 for IPv6 native encoding), if the message is sent for a single group`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-33.1	draft-ietf-pim-sm-v2-new-12.txt s4.10.5 p115 Join/Prune Message Format							
MUST	`Within one PIM Join/Prune message, all the Multicast Group Addresses, Joined Source addresses and Pruned Source addresses MUST be of the same address family.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-34.1	draft-ietf-pim-sm-v2-new-12.txt s4.10.5.1 p116 Group Set Source List Rules							

MUST	`(*,G) source list entries have the Source-Address set to the`
	`address of the RP for group G, the Source-Address Mask-Len set to the`
	`full length of the IP address and have both the WC and RPT bits of the`
	`Encoded-Source-Address set.`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-34.2	draft-ietf-pim-sm-v2-new-12.txt s4.10.5.1 p116-117 Group Set Source List Rules							

MUST	`(S,G) source list entries have the Source-Address set to the`
	`address of the source S, the Source-Address Mask-Len set to the full`
	`length of the IP address and have both the WC and RPT bits of the`
	`Encoded-Source-Address cleared.`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-34.3	draft-ietf-pim-sm-v2-new-12.txt s4.10.5.1 p115 Group Set Source List Rules							

MUST	`The wildcard group set is represented by the entire multicast range`
	`- the beginning of the multicast address range in the group address`
	`field and the prefix length of the multicast address range in the`
	`mask length field of the Multicast Group Address,e.g ff00::/8 for IPv6.`
	`(This test is for IPv6)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-35.1	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							

MUST	`Source specific asserts are sent by routers forwarding`
	`a specific source on the shortest-path tree(SPT bit is TRUE).`
	`(S,G) Asserts have the Group-Address field set to the group G`
	`and Source-Address field set to source S`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-35.2 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							
	`(S,G) Asserts have RPT-bit set to 0`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-35.3 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							

	`Group specific asserts are sent by routers forwarding data for`
	`the group and source(s) under contention on the shared tree.`
	`(*,G) Asserts have the Group-Address field set to the group G`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-35.4 **MAY**	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							

	`For data triggered Asserts the Source-Address field MAY be`
	`set to the IP source address of the data packet that triggered`
	`the Assert and is set to INADDR_ANY otherwise`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-35.5 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							
	`(*,G) Asserts have RPT-bit set to 1`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-35.6	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							
MUST	`Assert message contains metric preference value lookup.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-35.7	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							
MUST	`Assert message contains metric value lookup.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-35.8	draft-ietf-pim-sm-v2-new-12.txt s4.10.6 p120 Assert Message Format							
MUST	`When an assert is sent for a (*,G) entry, the first bit in the metric preference (the RPT-bit) is set to 1`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-36.1	draft-ietf-pim-sm-v2-new-12.txt s4.12 p124 Timer Values							
MUST	`Hello Timer (HT(I)). This timer is used for Periodic interval for Hello messages.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-36.2	draft-ietf-pim-sm-v2-new-12.txt s4.5.2 p48 Receiving (*,G) Join/Prune Messages							
MUST	In Join(J) state if the Expiry Timer for the (*,G) downstream state machine on interface I expires. The (*,G) downstream state machine on interface I transitions to the NoInfo state.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-36.3	draft-ietf-pim-sm-v2-new-12.txt s4.12 p125 Timer Values							
MUST	Assert Timer (AT(*,G,I), AT(S,G,I)). This timer is used for period after last assert before assert state is timed out. Default: 180 secs.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-36.4	draft-ietf-pim-sm-v2-new-12.txt s4.12 p126 Timer Values							
MUST	Upstream Join Timer (JT(*,*,RP), JT(*,G), JT(S,G)). This timer is used for period between Join/Prune messages. Default: 60 seconds							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-36.5	draft-ietf-pim-sm-v2-new-12.txt s4.12 p126 Timer Values							
MUST	Upstream Join Timer (JT(*,*,RP), JT(*,G), JT(S,G)). Suppression period when someone else sends a J/P message so we don't need to do so. Value: rand(1.1 * t_periodic, 1.4 * t_periodic) when Suppression_Enabled(I) is true, 0 otherwise.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-36.7 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.12 p126 Timer Values							
	`Upstream Join Timer (JT(*,*,RP), JT(*,G), JT(S,G)). This timer is used for period between Join/Prune messages (Here JT(S,G)) is tested`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-36.8 **MUST**	draft-ietf-pim-sm-v2-new-12.txt s4.12 p127 Timer Values							
	`KeepAlive Timer (KAT(S,G)). This timer is the Period after last (S,G) data packet during which (S,G) Join state will be maintained even in the absence of (S,G) Join messages. Default : 210 seconds.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.1 **MUST**	draft-ietf-pim-sm-bsr-12.txt s1.2 p7 Protocol Overview							
	`BSMs are originated periodically to ensure consistency after failure restoration.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.2 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	`If Bootstrap Timer expires, and current state is ` + "`" + `P-BSR', the router goes to E-BSR state and after receiving a non-preferred BSM, it remains in the E-BSR state and originates a BSM that contains the BSR priority value of the included BSR & the address of the bootstrap router for the domain.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-41.3 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	`In E-BSR state and after receiving a preferred BSM, it goes to the C-BSR state & forward BSM; store RP-Set; set Bootstrap timer to BS_Timeout.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.4 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	`In P-BSR state and after receiving a preferred BSM, it goes to the C-BSR state & forward BSM; store RP-Set; set Bootstrap timer to BS_Timeout.`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.5 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	`In P-BSR state and after receiving a non-preferred BSM, it remains in the P-BSR state & forward BSM`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.6 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	`In C-BSR state and after receiving a preferred BSM, it remains in the C-BSR state & forward BSM; store RP-Set; set bootstrap timer to BS_Timeout`							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-41.7 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	In C-BSR state and after receiving a preferred BSM, it remains in the C-BSR state & forward BSM; store RP-Set; set bootstrap timer to BS_Timeout (Note: A Bootstrap message is also preferred if it is from the current BSR with a lower weight than the previous BSM it sent, provided that if the router is a Candidate BSR the current BSR still has a weight higher or equal than the router itself.)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.8 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	In C-BSR state and after receiving a non-preferred BSM, it goes to the P-BSR state & forward BSM; set bootstrap timer to &lt;BS_Rand_Override&gt; (Note:A Bootstrap message is received from the elected BSR, but the BSR Priority field in the received message has changed, so that now the currently elected BSR has lower weight that the router itself.)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.9 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	In C-BSR state when bootstrap timer expires, it goes to the P-BSR state & set bootstrap timer to BS_Rand_Override							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.10 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.1.1 p11 Per-Scope-Zone Candidate-BSR State Machine							
	In E-BSR state if the BS Timer expires the BSR originates BSM and set BS Timer to BS_Period							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-41.11	draft-ietf-pim-sm-bsr-12.txt s3.1.2 p13 Per-Scope-Zone State Machine for Non-Candidate-BSR Routers							
MUST	If the included BSR is not preferred over, and not equal to, the currently active BSR If the Bootstrap Timer has expired and the receiving router is not a C-BSR, the Bootstrap message is then forwarded (Note: Per-Scope-Zone State-machine for Non-Candidate-BSR Routers)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.12	draft-ietf-pim-sm-bsr-12.txt s3.1.2 p13 Per-Scope-Zone State Machine for Non-Candidate-BSR Routers							
MUST	The router knows the identity of the current BSR, and is using the RP-Set provided by that BSR. Only bootsrap messages from that BSR or from a C-BSR with higher weight than the current BSR will be accepted							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.13	NEGATIVE draft-ietf-pim-sm-bsr-12.txt s3.1.2 p13 Per-Scope-Zone State Machine for Non-Candidate-BSR Routers							
MUST	The router knows the identity of the current BSR, and is using the RP-Set provided by that BSR. Only bootsrap messages from that BSR or from a C-BSR with higher weight than the current BSR will be accepted							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.14	draft-ietf-pim-sm-bsr-12.txt s3.2 p19 Sending Candidate-RP-Advertisement Messages							
MUST	Every C-RP periodically unicasts a C-RP-Adv to the BSR... (Note: Here the unicast test is performed)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-41.15	draft-ietf-pim-sm-bsr-12.txt s3.2 p19 Sending Candidate-RP-Advertisemnt Message							
MUST	Every C-RP periodically unicasts a C-RP-Adv to the BSR... (Note: Here the periodic test is performed)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.16	draft-ietf-pim-sm-bsr-12.txt s3.2 p19 Sending Candidate-RP-Advertisement Messages							
SHOULD	C-RPs should by default send C-RP-Adv messages withthe Priority field set to 192.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.17	draft-ietf-pim-sm-bsr-12.txt s3.2 p19 Sending Candidate-RP-Advertisement Messages							
MUST	If the C-RP is a ZBR for an admin scope zone, then the Admin Scope Zone bit MUST be set in the C-RP-Adv messages it sends for that scope zone; otherwise this bit MUST NOT be set. (Note: Admin Scope Zone bit is unset)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.18	draft-ietf-pim-sm-bsr-12.txt s3.3 p21 Creating the RP-Set at the BSR							
MUST	For each RP-address, the "RP-Holdtime" field is set to the Holdtime from the C-RP-Set, subject to the constraint that it MUST be larger than BS_Period and SHOULD be larger than 2.5 times BS_Period to allow for some Bootstrap messages getting lost.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-41.19 **SHOULD**	draft-ietf-pim-sm-bsr-12.txt s3.3 p21 Creating the RP-Set at the BSR							
	For each RP-address, the "RP-Holdtime" field is set to the Holdtime from the C-RP-Set, subject to the constraint that it MUST be larger than BS_Period and SHOULD be larger than 2.5 times BS_Period to allow for some Bootstrap messages getting lost. (Note: Here we test the SHOULD part "...SHOULD be larger than 2.5 times BS_Period")							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.20 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.3 p21 Creating the RP-Set at the BSR							
	There MUST however be a minimum of BS_Min_Interval between each time a BSM is sent.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.21 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.4 p23 Forwarding Bootstrap Messages							
	One is that a bootstrap message is not forwarded if its No-Forward bit is set, ...							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.22 **MUST**	draft-ietf-pim-sm-bsr-12.txt s3.4 p23 Forwarding Bootstrap Messages							
	When a Bootstrap message is forwarded, it is forwarded out of every multicast-capable interface which has PIM neighbors (including the one over which the message was received).							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-41.23 **MAY**	draft-ietf-pim-sm-bsr-12.txt s3.5 p24 Bootstrap Messages to New and Rebooting Routers							
	one router on the LAN sends a stored copy of the Bootstrap message for each admin scope zone to the new or rebooting router...This message SHOULD be sent as a No-Forward Bootstrap message ... For backwards compatibility, this message MAY instead or in addition be sent as a Unicast Bootstrap message,... (Note: Here ANVL checks that whether the Bootstrap MSG send by DUT has Multicast or Unicast destination. If the destination is Multicast then it should be No-Forward Bootstrap message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.24 **MUST**	NEGATIVE draft-ietf-pim-sm-bsr-12.txt s3.5 p24 Bootstrap Messages to New and Rebooting Routers							
	To allow new or rebooting routers to learn the RP-Set quickly, when a Hello message is received from a new neighbor, or a Hello message with a new GenID is received from an existing neighbor, one router on the LAN sends a stored copy of the Bootstrap message for each admin scope zone to the new or rebooting router.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.26 **MUST**	draft-ietf-pim-sm-bsr-12.txt s4 p25 Message Formats							
	Usually, Bootstrap messages are multicast with TTL 1 to the ALL-PIM-ROUTERS group, ... (Note: Here DUT originates the Bootstrap Message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.27 **MUST**	draft-ietf-pim-sm-bsr-12.txt s4 p25 Message Formats							
	Usually, Bootstrap messages are multicast with TTL 1 to the ALL-PIM-ROUTERS group, ... (Note: Here DUT forwards the Bootstrap Message)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-41.28 **MUST**	draft-ietf-pim-sm-bsr-12.txt s4 p25 Message Formats							

	`Usually, Bootstrap messages are multicast with TTL 1 to the`
	`ALL-PIM-ROUTERS group, ...`
	`(Note: here we check IP TTL value)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.29 **MUST**	draft-ietf-pim-sm-bsr-12.txt s4 p25 Message Formats							

	`Usually, Bootstrap messages are multicast with TTL 1 to the`
	`ALL-PIM-ROUTERS group, but in some circumstances (described`
	`in section 3.5.2) Bootstrap messages are unicast to a specific`
	`PIM neighbor.`
	`(Note: here we check IP TTL value for forwarded BSM)`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-41.30 **MAY**	draft-ietf-pim-sm-bsr-12.txt s4.1 p28 Bootstrap Message Format							

	`The length (in bits) of the mask to use in the hash function. For IPv6`
	`we recommend a value of 126.`							

	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-41.31 **MUST**	draft-ietf-pim-sm-bsr-12.txt s4.2 p32 Candidate-RP-Advertisement Message Format							
	`` C-RPs MUST NOT send C-RP-Adv messages with a Prefix Count of `0'. ``							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							

	Release 8.4	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x	Release x.x.x
PIM-SMV6-42.1	draft-ietf-pim-sm-bsr-12.txt s3.6 p25 Receiving and Using the RP-Set							
MUST	If a mapping is not already part of the RP-Set, it is added to the RP-Set and the associated Group-to-RP mapping Expiry Timer (GET) is initialized to the holdtime from the Bootstrap message. Its priority is set to the Priority from the Bootstrap message.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: FAIL							
	Free BSD 12.0 untested							
PIM-SMV6-42.2	draft-ietf-pim-sm-bsr-12.txt s3.6 p25 Receiving and Using the RP-Set							
MUST	If a mapping is already part of the RP-Set, it is updated with the Priority from the Bootstrap message and its associated GET is reset to the holdtime from the Bootstrap message.							
	Free BSD 10.3 untested							
	Ubuntu 18.04: inconclusive							
	Free BSD 12.0 untested							
PIM-SMV6-42.3	draft-ietf-pim-sm-bsr-12.txt s3.6 p25 Receiving and Using the RP-Set							
MUST	If a mapping is not already part of the RP-Set, it is added to the RP-Set and the associated Group-to-RP mapping Expiry Timer (GET) is initialized to the holdtime from the Bootstrap message. Its priority is set to the Priority from the Bootstrap message. (Note: This test is for rp-priority)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							
PIM-SMV6-42.4	draft-ietf-pim-sm-bsr-12.txt s3.6 p25 Receiving and Using the RP-Set							
MUST	If a mapping is already part of the RP-Set, it is updated with the Priority from the Bootstrap message and its associated GET is reset to the holdtime from the Bootstrap message. (Note: This test is for rp-priority)							
	Free BSD 10.3 untested							
	Ubuntu 18.04: pass							
	Free BSD 12.0 untested							