Network Working Group
Request for Comments: 2174
Category: Informational

K. Murakami
M. Maruyama
NTT Laboratories

June 1997

A MAPOS version 1 Extension - Switch-Switch Protocol

Status of this Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Authors' Note

This memo documents a MAPOS (Multiple Access Protocol over SONET/SDH) version 1 extension, Switch Switch Protocol which provides dynamic routing for unicast, broadcast, and multicast. This document is NOT the product of an IETF working group nor is it a standards track document. It has not necessarily benefited from the widespread and in depth community review that standards track documents receive.

Abstract

This document describes a MAPOS version 1 extension, SSP (Switch Switch Protocol). MAPOS is a multiple access protocol for transmission of network-protocol packets, encapsulated in High-Level Data Link Control (HDLC) frames, over SONET/SDH. In MAPOS network, a SONET switch provides the multiple access capability to end nodes. SSP is a protocol of Distance Vector family and provides unicast and broadcast/multicast routing for multiple SONET switch environment.

1. Introduction

This document describes an extension to MAPOS version 1, Switch Switch Protocol, for routing both unicast and broadcast/multicast frames. MAPOS[1], Multiple Access Protocol over SONET (Synchronous Optical Network) / SDH (Synchronous Digital Hierarchy) [2][3][4][5], is a link layer protocol for transmission of HDLC frames over SONET/SDH. A SONET switch provides the multiple access capability to each node. SSP is a dynamic routing protocol designed for an environment where a MAPOS network segment spans over multiple switches. It is a protocol of Distance Vector family. It provides both unicast and broadcast/multicast routing. First, this document describes the outline of SSP. Next, it explains unicast and broadcast/multicast routing algorithms. Then, it describes the SSP protocol in detail.

2. Constraints in Designing SSP

SSP is a unified routing protocol supporting both unicast and broadcast/multicast. The former and the latter are based on the Distance Vector [6][7] and the spanning tree[8] algorithm, respectively. In MAPOS version 1, a small number of switches is assumed in a segment. Thus, unlike DVMRP(Distance Vector Multicast Routing Protocol)[8], TRPB(Truncated Reverse Path Broadcasting) is not supported for simplicity. This means that multicast frames are treated just the same as broadcast frames and are delivered to every node.

In MAPOS version 1, there are two constraints regarding design of the broadcast/multicast routing algorithm;

(1) there is no source address field in MAPOS HDLC frames

(2) there is no TTL(Time To Live) field in MAPOS HDLC frames to prevent forwarding loop.

To cope with the first issue, VRPB(Virtual Reverse Path Broadcast) algorithm is introduced. In VRPB, all broadcast and multicast frames are assumed to be generated by a node under a specific switch called VSS(Virtual Source Switch). VSS is the switch which has the smallest switch number in a MAPOS network. Each switch determine its place in the spanning tree rooted from VSS independently. Whenever a switch receives a broadcast/multicast frame, it forwards the frame to all upstream and downstream switches except for the one which has sent the frame to the local switch.

To cope with the second issue, the forward delay timer is introduced. Even if a switch finds a new VSS, it suspends forwarding for a time period. This timer ensures that all the switches have a consistent routing information and that they are synchronized after a topology change.

3. Unicast Routing in SSP

This section describes the address structure of MAPOS version 1 and the SSP unicast routing based on it.

3.1 Address Structure of MAPOS version 1

In a multiple switch environment, a node address consists of the switch number and the port number to which the node is connected. As shown in Figure 1, the address length is 8 bits and the LSB is always 1, which indicates the end of the address field. A MSB of 0 indicates a unicast address. The switch and the port number fields are variable-length. In this document, a unicast address is represented as "0 <switch-number> <port number>". Note that a port number includes EA bit.

MSB of 1 indicates multicast or broadcast. In the case of broadcast, the address field contains all 1s (0xff in hex). In the case of multicast, the remaining bits indicate a group address. The switch number field is variable-length. A multicast address is represented as "1 <group address>".

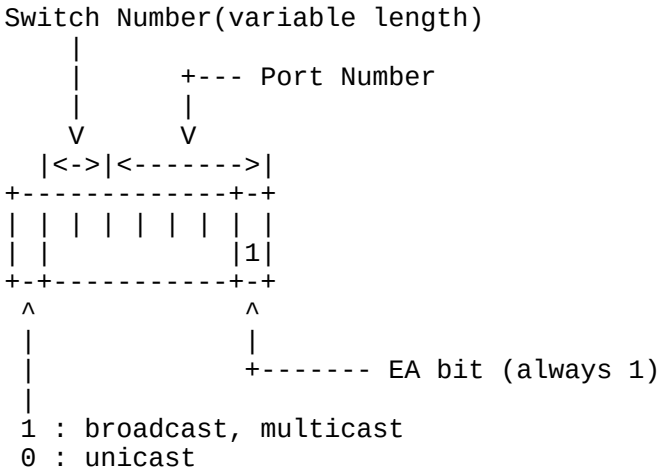

Figure 1 Address Format

Figure 2 shows an example of a SONET LAN that consists of three switches. In this configuration, two bits of a node address are used to indicate the switch number. Node N1 is connected to port 0x03(000011 in binary) of the switch S2 numbered 0x2. Thus, the node address is 01000011 in binary. Node N4 has an address 01101001 in binary since the connected switch number is 0x3 and the port number is 0x09.

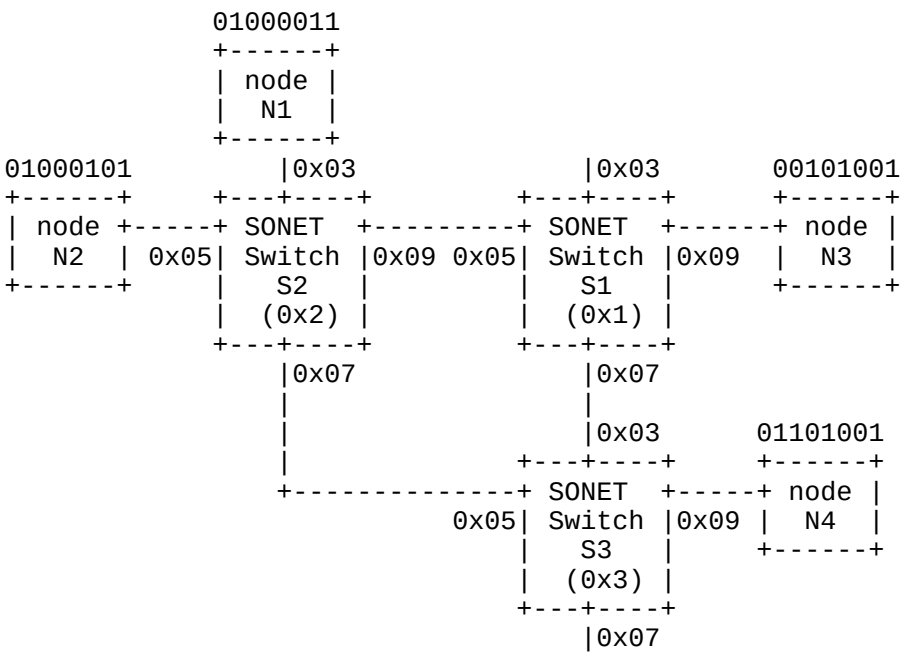

Figure 2 Multiple SONET Switch Environment

3.2 Forwarding Unicast Frames

Unicast frames are forwarded along the shortest path. For example, a frame from node N4 destined to N1 is forwarded by switch S3 and S2. These SONET switches forwards an HDLC frame based on the destination switch number contained in the destination address.

Each switch keeps a routing table with entries for possible destination switches. An entry contains the subnet mask, the next hop to the adjacent switch along the shortest path to the destination, the metric measuring the total distance to the destination, and other parameters associated with the entry such as timers. For example, the routing table in switch S1 will be as shown in Table 1. The metric value 1 means that the destination switch is an adjacent switch. The value 16 means that it is unreachable. Although the values between 17 and 31 also mean unreachable, they are special values utilized for split horizon with poisoned reverse [8].

destination switch	subnet mask	next hop port	metric	other parameters
01000000	11100000	00000101	1	
01100000	11100000	00000111	1	

Table 1 An Example of a Routing Table

When a switch receives a unicast frame, it extracts the switch number from the destination address. If it equals to the local switch number, the frame is sent to the local node through the port specified in the destination address. Otherwise, the switch looks up its routing table for a matching destination switch number by masking the destination address with the corresponding subnet mask. If a matching entry is found, the frame is sent to an adjacent switch through the next hop port in the entry. Otherwise, it is silently discarded or sent to the control processor for its error processing.

3.4 Protocol Overview

This subsection describes an overview of the unicast routing protocol and its algorithm.

3.4.1 Route Exchange

SSP is a distance vector protocol to establish and maintain the routing table. In SSP, each switch sends a routing update message to every adjacent switches every FULL_UPDATE_TIME (10 seconds by default). The update message is a copy of the routing table, that is, routes.

When a switch receives an update message from an adjacent switch through a port, it adds the cost associated with the port, usually 1, to every metric value in the message. The result is a set of new metrics from the receiving switch to the destination switches. Next, it compares the new metrics with those of the corresponding entries in the existing routing table. A smaller metric means a better route. Thus, if the new metric is smaller than the existing one, the entry is updated with the new metric and next hop. The next hop is the port from which the update message was received. Otherwise, the entry is left unchanged. If the existing next hop is the same as the new one, the metric is updated regardless of the metric value. If no corresponding route is found, a new route entry is created.

3.4.2 Route Expiration

Assume a route to R is advertised by a neighboring switch S. If no update message has been received from switch S for the period FULL_UPDATE_TIME * 3 (30 seconds by default) or the route is advertised with metric 16 by switch S, the route to R is marked as unreachable by setting its metric to 16. In other words, the route to R is kept advertised even if the route is not refreshed up-to 30 seconds.

To process this, each routing table entry has an EXPIRATION_TIMER (30 seconds by default, that is, FULL_UPDATE_TIME *3). If another switch advertises a route to R, it replaces the unreachable route. Even if a route is marked unreachable, the entry is kept in the routing table for the period of FULL_UPDATE_TIME * 3. This enables the switch to notify its neighbors of the unreachable route by sending update messages with metric 16. To process this, each routing table entry has a garbage collection timer GC_TIMER (30 seconds by default). The entry is deleted on expiration of the timer. Figure 3 shows this transition.

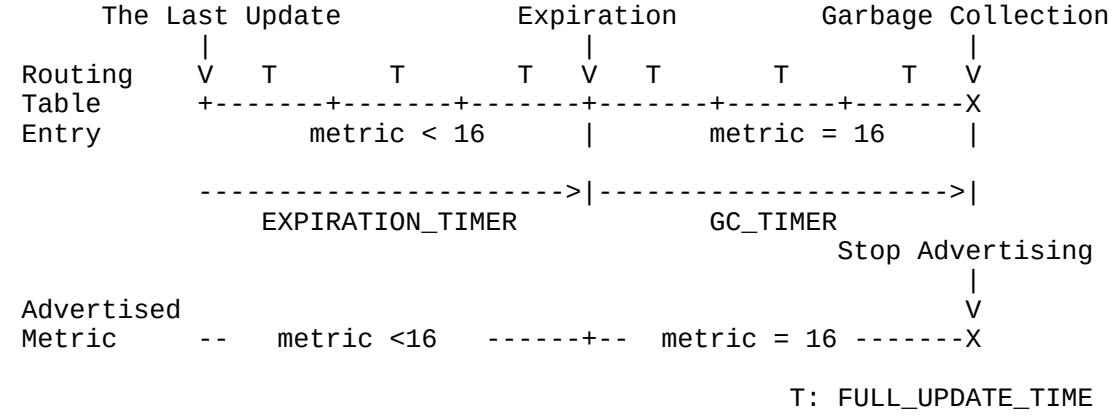

Figure 3. Route Expiration

3.4.3 Slow Convergence Prevention

To prevent slow convergence of routing information, two techniques, split horizon with poisoned reverse, and triggered update are employed.

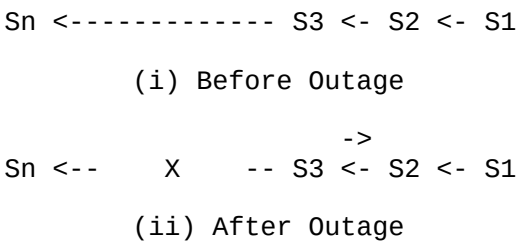

Figure 4 An Example of Slow Convergence

Figure 4 shows an example of slow convergence[6]. In (i), three switches, S1, S2, and S3, are assumed to have a route to Sn. In (ii), the connection to Sn has disappeared because of an outage, but S2 continue to advertise the route since there is no means for S2 to detect the outage immediately and it has the route to Sn in its routing table. Thus, S3 misunderstand that S2 has the best route to Sn and S2 is the next hop. This results in a transitive loop between S2 and S3. S2 and S3 increments the metric of the route to Sn every time they advertise the route and the loop continues until the metric reaches 16. To suppress the slow convergence problem, split horizon with poisoned reverse is used.

In split horizon with poisoned reverse, a route is advertised as unreachable to the next hop. The metric is the received metric value plus 16. For example, in Figure 4, S2 advertises the route to Sn with the metric unreachable only to S3. Thus, S3 never considers that S2 is the next hop to Sn. This ensures fast convergence on disappearance of a route.

Another technique, triggered update, forces a switch to send an immediate update instead of waiting for the next periodic update when a switch detects a local port failure, or when it receives a message that a route has become unreachable, or that its metric has increased. This makes the convergence faster.

4. Broadcast/multicast Routing in SSP

This section explains VRPB algorithm and the outline of broadcast/multicast routing protocol.

4.1 Virtual Reverse Path Broadcast/Multicast Algorithm

SSP provides broadcast/multicast routing based on a spanning tree algorithm. As described in Section 2, the routing is based on the VRPB(Virtual Reverse Path Broadcast) algorithm. In VRPB, each switch assumes that all broadcast and multicast frames are generated by a specific switch, VSS(Virtual Source Switch). Thus, unlike DVMRP, a MAPOS network has only one spanning tree at any given time.

The frames are forwarded along the reverse path by computing the shortest path from the VSS to all possible recipients. VSS is the switch which has the lowest switch number in the network. Because the routing table contains all the unicast destination addresses including the switch numbers, each switch can identify the VSS independently by searching for the smallest switch number in its unicast routing table.

In Figure 2, switch S1 is the VSS. Each switch determines its place in the spanning tree, relative to the VSS, and which of its ports are on the shortest path tree. Thus, the spanning tree is as shown in Figure 5. Except for the VSS, each switch has one upstream port and zero or more downstream ports. VSS have no upstream port, since it is the root of the spanning tree. In Figure 2. switch S2's upstream port is port 0x09 and it has no downstream port.

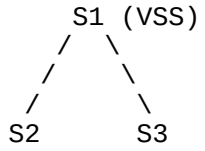

Figure 5 VRPB Spanning Tree

When a switch receives a broadcast/multicast frame, it forwards the frame to all of the upstream switch, the downstream switches, and the directly connected nodes. However, it does not forward to the switch which sent the frame to it. For that purpose, a bit mapped broadcast/multicast routing table may be employed. The broadcast/multicast routing process marks all the bits corresponding to the ports to which frames should be forwarded. The forwarding process refers to it and broadcasts a frame to all the ports with its corresponding bit marked.

4.2 Forwarding Broadcast/multicast Frames

When a switch forwards a broadcast/multicast frame, (1) it first decides the VSS by referring to its unicast routing table. Then, (2) it refers to its broadcast/multicast routing table corresponding to

the VSS. A cache may be used to reduce the search overhead. (3) Based on the routing table, the switch forwards the frame.

Figure 6 shows an example of S2's broadcast/multicast routing table for the VSS S1. It is a bit map table and each bit corresponds to a port. The value 1 indicates that frames should be forwarded to a node or a switch through the port. If no bit is marked, the frame is silently discarded. In the example of Figure 6, port 0x09 is the upstream port to its VSS, that is, S1. Other ports, ports 0x05 and 0x03 are path to N2 and N1 nodes, respectively.

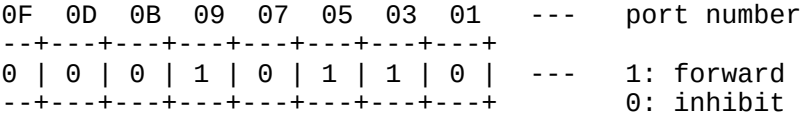

Figure 6 Broadcast/Multicast Routing Table of S2

4.3 Forwarding Path Examples

Assume that a broadcast frame is generated by N2 in Figure 2. The frame is received by S2.

Then, S2 passes it to all the connected nodes except for the source N2. That is, only to N1. At the same time, it also forwards the frame to all its upstream and downstream switches. Since S2 has no downstream switch, S2 forwards the frame to S1 though its upstream port 0x09.

S1 is the VSS and it passes the frame to all the local nodes, that is, only to N3. Since it has no upstream switch and S2 is the switch which sent the frame to S1, the frame is eventually forwarded only to a downstream switch S3.

S3 passes the frame to its local node, N4. Since S3 has only an upstream and the frame was received through that port, S3 does not forward the frame to any switch.

The resulting path is shown in Figure 7. Although this is not the optimal path, VRPB ,at least, ensures that broadcast/multicast frames are delivered all the nodes without a loop. Figures 8 and 9 show the forwarding path for frames generated by a node under S3 and S4, respectively.

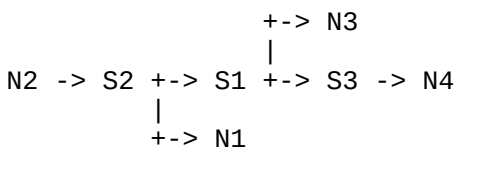

Figure 7 Forwarding Path from N2

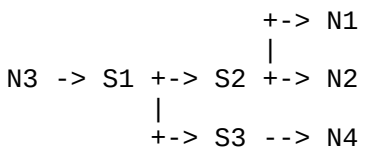

Figure 8 Forwarding Path from N3

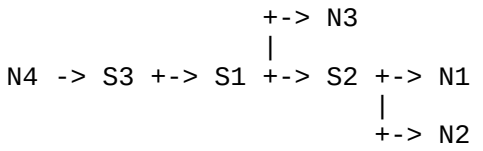

Figure 9 Forwarding Path from N4

4.4 Suppressing Routing Loop

To suppress transitive routing loop, forward delay is employed. A switch suspends broadcast/multicast forwarding for a period after a new VSS is found in the routing table. This prevents transitive routing loop by waiting for all the switches to have the same routing information and become synchronized. In addition to controlling sending of frames by forward delay, another mechanism is employed to prevent transitive routing loop by controlling reception of frames. That is, broadcast/multicast frames received through ports other than the upstream and downstream ports are discarded.

4.5 Upstream Switch Discovery

The upstream port is determined by the shortest reverse path to the VSS. It is identified by referring to the next hop port of the route to VSS in the local unicast routing table. When a new next hop to the VSS is discovered, the bit corresponding to the old next hop port is cleared, and the bit corresponding to the new one is marked as the upstream port in the broadcast/multicast routing table.

4.6 Downstream Switch Discovery

To determine the downstream ports, split horizon with poisoned reverse is employed. When a switch receives a route with a metric poisoned by split horizon processing through a port as described in Section 3.4.3, the port is considered to be a downstream port. In Figure 2, S1 is the VSS and the route information is sent back from S2 to S1 with metric unreachable based on the split horizon with poisoned reverse. Thus, S1 knows that S2 is one of its downstreams.

4.7 Downstream Port Expiration

When a poison reversed packet is newly received from a port, the local switch knows that a new downstream switch has appeared. Then, it marks the bit corresponding to the port and starts FORWARD_DELAY_TIMER (30second by default, that is, FULL_UPDATE_TIME * 3) for the port. The forwarding of broadcast/multicast frames to the port is prohibited until the timer expires. Every time the local switch receives a poison reversed packet through a port, it initializes PORT_EXPIRATION_TIMER(30 seconds by default, that is, FULL_UPDATE_TIME *3) corresponding to the port. A continuous loss of poison reversed packets or a failure of downstream port results in expiration of PORT_EXPIRATION_TIMER, and the corresponding bit is cleared.

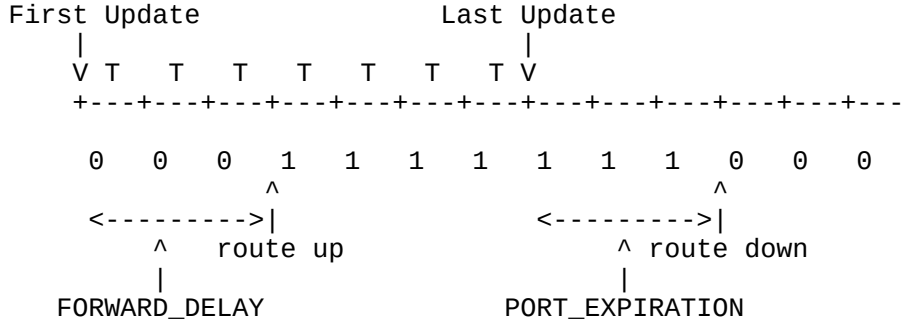

Figure 10. Port Expiration

When a downstream switch discovers another best path to the VSS or a new VSS, it stops split horizon with poison reverse and sends ordinary update messages. Whenever the local switch receives an ordinary update message from its downstream switch, it SHOULD immediately clear the corresponding bit in the routing table and stop forwarding of broadcast/multicast frames.

4.8 Node Discovery

When a NSP[9] packet, requesting a node address from a port, is received, the local switch considers that a new node is connected, and marks the corresponding bit in the broadcast/multicast routing table. When the local switch detects that the port went down as described in [9], it clear the corresponding bit.

4.9 Invalidating The Broadcast/multicast Routing Table

When a new VSS is discovered or when the VSS becomes unreachable, the entire broadcast/multicast routing table is invalidated. That is, a change of upstream port affects the entire broadcast/multicast routing. However, a change of a downstream port does not affect forwarding to other downstream ports, its upstream port, and nodes.

5. Detailed Protocol Operation

This section explains SSP packet format and protocol processing in detail.

5.1 Packet Format

This subsection describes the packet encapsulation in HDLC frame and the packet format.

5.1.1 Packet Format and Its Encapsulation

SSP packet format is designed based on RIP[6] and its successor, RIP2 [7]. Figure 11 shows the packet format. A SSP packet is encapsulated in the information field of a MAPOS HDLC frame. The HDLC protocol field of SSP is 0xFE05 in hex as defined by the "MAPOS Version 1 Assigned Numbers" [10]. The packet is sent encapsulated in a unicast packet with the destination address 0000 0001, which indicates the control processor of an adjacent switch.

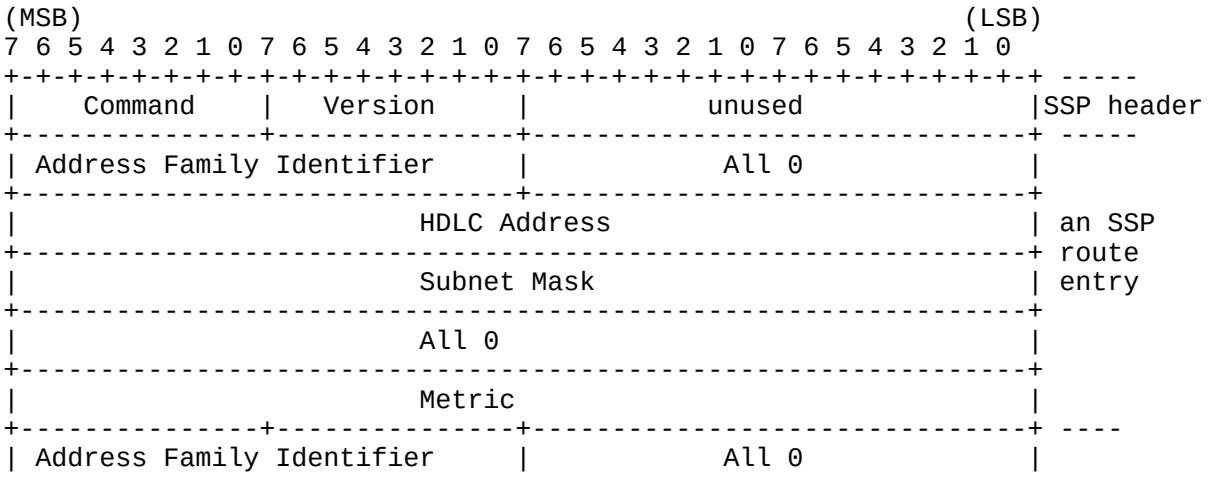

Figure 11 SSP packet format

The maximum packet size is 512 octet. The first four octets is the SSP header. The remainder of the message is composed of 1 - 25 route entries. Each entry is 20 octets long.

5.1.2 SSP Header

SSP header consists of a command field and a version field. The command field is one octet long and holds one of the following values;

1 - request	A request to send all or part of SSP routing table.
2 - response	A message containing all, or a part of the sender's SSP routing table. This message may be sent in response to a request, or it may be an update message generated by the sender.

The Version field indicates the version of SSP being used. The current version number is 1.

5.1.3 SSP Route Entries

Each entry has an address family identifier. It indicates an attribute of the entry. SSP routing protocol uses 2 as its identifier by default. The identifier 0 indicates unspecified. This value is used when a switch requests other switches to send the entire SSP routing table. A recipient of the message SHOULD ignore all entries with unknown value.

The HDLC address is a destination address. It may be a switch address or a node address. The subsequent subnet mask is applied to the HDLC address to yield the switch number portion. The field is 4 octet long and the address is placed in the least significant position.

Metric indicates the distance to the destination node. That is, how many switches a message must go through en route to the destination node. The metric field must contain a value between 1 and 31. The metric of 16 indicates that the destination is not reachable and is ignored by recipients. The values between 17 and 31 are utilized for poisoned reverse with split horizon and also means unreachable. The metric 0 indicates the local switch itself.

5.2 Routing Table

Every switch has an SSP routing table. The table is a collection of route entries - one for every destination. An entry consists of the following information;

(1) destination : A unicast destination address.

(2) subnet mask : A mask to extract the switch address by applying bitwise AND with the destination address

(3) next hop port : The local port number connected to the adjacent switch along the path to the destination.

(4) metric : Distance to the destination node. The metric of an adjacent switch is 1 and that of local switch is 0.

(5) timers for unicast routing : Timers associated with unicast routing such as EXPIRATION_TIMER and GC_TIMER.

(6) flags : Various flags associated with the route such as route change flag to indicate that the route has changed recently or it has timed out.

(7) bit map routing table for broadcast/multicast : Each bit corresponding to the port to an upstream or a downstream switch of the spanning tree is marked in addition to the ports to end nodes. Broadcast/multicast frames are forwarded only through those ports with their corresponding bit set. Since only one spanning tree exists at a time in a network, each route entry does not necessarily have to have this field.

(8) timers for broadcast/multicast routing : Timers associated with broadcast/multicast routing such as FORWARD_DELAY_TIMER and PORT_EXPIRATION_TIMER. These timers are prepared for each bit of broadcast/multicast routing table.

5.3 Sending Routing Messages

5.3.1 Packet Construction

Because of the split horizon with poisoned reverse, a routing message differs depending on the adjacent switch to which the message is being sent. The upstream switch of a route, that is next hop, receives a message which contains the corresponding route with a metric between 17 and 31. Switches that are not the upstream switch of any route receive the same message. Here, we assume that a packet for a routing message is constructed for an adjacent switch which is connected through the local port N.

First, set the version field to 1, the current SSP version. Then, set the command to "response". Set other fields which are supposed to be zero to zero. Next, start filling in entries.

To fill in the entries, perform the following for each route. The destination HDLC address, netmask, and its metric are put into the entry in the packet. Routes must be included in the packet even if their metrics are unreachable(16). If the next hop port is N, 16 is added to the metric for split horizon with poisoned reverse.

Recall that the maximum packet size is 512 bytes. When there is no more space in a packet, send the current message and start a new one. If a triggered update is being generated, only entries whose route change flags are set need be included.

5.3.2 Sending update

Sending update may be triggered in any of the following ways;

(1) Initial Update

When a switch first comes up, it SHOULD send to all adjacent switches a request asking for their entire routing tables. The destination address is 00000001. When a port comes on-line, the request packet is sent to the port. The packet, requesting the entire routing table, MUST have at least an entry with the address family identifier 0 meaning unspecified.

When a switch receives a request packet, it first checks the version number of the SSP header. If it is not 1, the packet is silently

discarded. Otherwise, the address family identifier is examined. If the value is 0, the entire SSP routing table is returned in one or more response packets destined to 00000001. Otherwise, the request is silently discarded. Although the original RIP specification defines the partial routing table request, SSP routing protocol omits it for the sake of simplicity.

(2) Periodic Update

Every switch participating in the routing process sends an update message (response message) to all its neighbor switches once every FULL_UPDATE_TIME (10 seconds). For the periodic update, a response packet(s) is used. The destination address is always 00000001. An update message contains the entire SSP routing table. The maximum packet size is 512byte. Thus, an update message may require several packets to be packed.

(3) Triggered Update

When a route in the unicast routing table is changed or a local port goes down, the switch advertises a triggered update packet without waiting for the full update time. The difference between triggered update and the other update is that triggered updates do not have to include the entire routing table. Only changed entries should be included. Triggered update may be suppressed if a regular periodic update is due.

Note that when a route is advertised as unreachable (metric 16) by an adjacent switch, update process is triggered as well as expiration of the route in the local switch.

(4) On Termination

When a switch goes down, it is desirable to advertise all the routes with metric 16, that is, unreachable.

5.4 Receiving Routing Messages

When a switch receives an update, it first checks the version number. If it is not 1, the update packet is silently discarded. Otherwise, it processes the entries in it one by one.

For each entry, the address family identifier is checked. If it is not 2, the entry is ignored. Otherwise, the metric is checked. The value should be between 0 and 31. An entry with illegal metric is ignored. Next, the HDLC address and the subnet mask is checked. An entry with an invalid address such as broadcast is ignored. If the entry passed all these validation checks, it is processed according to the following steps;

Step 1 - Process Poisoned Reverse

If the metric value is between 0 and 16, it is an unicast information. Go ahead to Step 2.

If the metric value is between 17 and 31, it indicates poisoned reverse, that the local switch has been chosen as the next hop for the route. However, if the corresponding entry is not included in the current routing table or the message is from a port connected to its upstream switch, the message is illegal -- ignore it and return to Step 1 to process the next entry. Otherwise,

(1) Initialize the PORT_EXPIRATION_TIMER corresponding to the downstream port.
(2) Operate the FORWARD_DELAY_TIMER as follows;
    (2-1) If the broadcast/multicast forwarding was already enabled, go to (3).
    (2-2) If the FORWARD_DELAY_TIMER corresponding to the downstream port was already started, increment the timer. If the timer expires, mark the bit in the broadcast/multicast routing table corresponding to the port and stop the timer.
    (2-2) Otherwise, start the FORWARD_DELAY_TIMER.
(3) Return to Step 1 to process the next entry.

Step 2 - Process Unicast Routing Information

First, add the cost associated with the link, usually 1, to the metric. If the result is greater than 16, 16 is used. Then, look up the unicast routing table for the corresponding entry. There are two cases.

Case 1 no corresponding entry is found

If the new metric is 16, return to step 1 to process the next entry. Otherwise,
(1) Create a new route entry in the routing table
(2) Initialize EXPIRATION_TIMER and GC_TIMER

(3) The port corresponding to the new route is the next_hop port for the route. Thus, mark the bit in the broadcast/multicast routing table corresponding to the new next_hop port and start FORWARD_DELAY_TIMER. If this new route is for the switch with the minimum switch number, select it as the VSS and use its broadcast/multicast routing table. (See NOTE 1.)
(4) Set the route change flag and invoke triggered update process
(5) Return to step 1 to process the next entry.

[NOTE 1]
There are two implementations;
(1) Prepare a spanning tree for each route and use only one corresponding to the current VSS. In this case, each unicast route entry has a broadcast/unicast routing table.
(2) Prepare only one spanning tree corresponding to the current VSS. In this case, a switch has only one broadcast/multicast routing table.
In this document, the former is assumed.

Case 2. A corresponding entry is found

In this case, the update message is processed differently according to the new metric value.

(a) new_metric < 16 & new_metric > current_metric

(1)If and only if the update is from the same port(next_hop port) as the existing one,
(1-1) Update the entry
(1-2) Initialize EXPIRATION_TIMER and GC_TIMER

(2) If the corresponding bit to the port, which the update message is received, is marked in the broadcast/multicast routing table, clear the bit.
(3) Return to Step 1 and process the next entry.

(b) new_metric < 16 & new_metric < current_metric

(1) Update the entry and clear the bit in the broadcast/multicast routing table corresponding to the old next_hop port.
(2) Initialize EXPIRATION_TIMER, GC_TIMER, and PORT_EXPIRATION_TIMER for the new next_hop port.
(3) Mark a bit in the broadcast/multicast routing table corresponding to the new next_hop port and start FORWARD_DELAY_TIMER.

(4) Set the route change flag and invoke triggered update with poisoned reverse for the new next_hop.
(5) Return to Step 1 to process the next entry.

(c) new_metric < 16 & new_metric = current_metric

If a new route with the same metric value as the existing routing table entry is received, use the old one as follows;

(1) If the new next hop is equal to the current one, initialize EXPIRATION_TIMER and GC_TIMER. Otherwise, ignore this update.
(2) If the bit corresponding to the port, from which the update message was received, is marked in the broadcast/multicast routing table, clear the bit.
(3) Return to Step 1 to process the next entry.

(d) the new metric = 16 & the new next hop = the current one

If the current metric is not equal to 16, this is a new unreachable information. Then,
(1) Update the entry and clear the bit in the broadcast/multicast routing table corresponding to the old next_hop port.
(2) If this route is for the current VSS, select a new VSS in the valid routing table entries. Valid means that the destination is reachable.
(3) Set the route change flag and invoke triggered update process to notify the unreachable route.
Otherwise,
do nothing and return to Step 1 to process the next entry.

(e) the new metric = 16 & the new next hop /= the current one

(1) If the bit corresponding to the port, from which the update message was received, is marked in the broadcast/multicast routing table, clear the bit.
(2) Return to Step 1 to process the next entry.

5.5 Timers

The timer routine increments the following timers and executes its associated process on their expiration.

(1) EXPIRATION_TIMER and GC_TIMER

The EXPIRATION_TIMERs and GC_TIMERs of each entry in the unicast routing table are incremented every FULL_UPDATE_TIME (10 seconds by default). When a EXPIRATION_TIMER expires, the metric is changed to unreachable(16), update process is triggered, and GC_TIMER is started. When a GC_TIMER expires, the entry is deleted from the local routing table. EXPIRATION_TIMER and GC_TIMER are cleared every time a switch receives a routing update.

(2) FORWARD_DELAY_TIMER

FORWARD_DELAY_TIMER is completely handled in the receive process and has no relation to the timer routine.

(3) PORT_EXPIRATION_TIMER

PORT_EXPIRATION_TIMERs associated with each bit in the broadcast/multicast routing table are incremented every FULL_UPDATE_TIME (10 seconds by default). When the timer expires, the corresponding downstream switch is considered to be down and the corresponding bit in the broadcast/multicast routing table is cleared. This timer is cleared by the receive process every time a poisoned reverse packet is received from the corresponding switch.

6. Further considerations on implementation

6.1 Port State

A switch assumes that every port is connected to a switch initially. Thus, it sends update packets to every port. When a node is connected to a port, the switch recognizes it by receiving an NSP request packet, and stops sending SSP packets to the port. Whenever a switch detects a connection failure such as loss of signal and out-of-synchronization, it should clear the internal state table corresponding of the port.

6.2 Half way connection problem

A port consists of two channels, transmit and receive. Although it is easy for a node or a switch to detect a receive channel failure, transmit channel failure may not be detected, causing half way connection. This results in a black hole.

Thus, whenever a switch receives a SSP update packet from a port, it SHOULD check the status of the corresponding transmit channel. SONET/SDH has a feedback mechanism for that purpose. The status of the local transmit channel received at the remote end can be sent back utilizing the overhead part, FEBE(Far End Block Error) and FERF(Far End Receive Failure), of the corresponding receive channel. If the signals indicates that the transmit channel has a problem, the SSP packet received from the remote end should be silently discarded. However, some SONET/SDH services do not provide path overhead transparency.

Although, SONET/SDH APS(Automatic Protection Switching) can be utilized to switch service from a failed line to a spare line, the function is out of scope of this protocol.

7. Security Considerations

Security issues are not discussed in this memo.

References

[1] Murakami, K. and M. Maruyama, "MAPOS - Multiple Access Protocol over SONET/SDH Version 1," RFC2171, June 1997.

[2] CCITT Recommendation G.707: Synchronous Digital Hierarchy Bit Rates, 1990.

[3] CCITT Recommendation G.708: Network Node Interface for Synchronous Digital Hierarchy, 1990.

[4] CCITT Recommendation G.709: Synchronous Multiplexing Structure, 1990.

[5] American National Standard for Telecommunications - Digital Hierarchy - Optical Interface Rates and Formats Specification, ANSI T1.105-1991.

[6] Hedrick, C., "Routing Information Protocol", STD 34, RFC 1058, Rutgers University, June 1988.

[7] Malkin, G., "RIP Version 2 - Carrying Additional Information ", RFC1723, Xylogics, Inc., November 1994.

[8] Pusateri, T., "Distance Vector Multicast Routing Protocol", September 1996, Work in Progress.

[9] Murakami, K. and M. Maruyama, "A MAPOS version 1 Extension - Node Switch Protocol," RFC2173, June 1997.

[10] Maruyama, M. and K. Murakami, "MAPOS Version 1 Assigned Numbers," RFC2172, June 1997.

Acknowledgements

The authors would like to acknowledge the contributions and thoughtful suggestions of John P. Mullaney, Clark Bremer, Masayuki Kobayashi, Paul Francis, Toshiaki Yoshida, Takahiro Sajima, and Satoru Yagi.

Authors' Address

Ken Murakami
NTT Software Laboratories
3-9-11, Midori-cho
Musashino-shi
Tokyo 180, Japan
E-mail: murakami@ntt-20.ecl.net

Mitsuru Maruyama
NTT Software Laboratories
3-9-11, Midori-cho
Musashino-shi
Tokyo 180, Japan
E-mail: mitsuru@ntt-20.ecl.net