Provider Provisioned VPN WG
Internet-draft
Expires: December 2002

N. Finn (Cisco)
M. Seaman (Consultant)
A. Smith (Consultant)
A. Romanow (Cisco)

Bridging and VPLS

draft-finn-ppvpn-bridging-vpls-00

Status of this Memo

Copyright Notice

Abstract

Layer 2 techniques based on IEEE 802.1Q bridges are in widespread use by Ethernet MAN Service Providers. It is possible to implement the data plane functionality of Service Provider Backbone as described in the framework draft [ANDERSSON] using bridges as the Provider Edge (PE) equipment. There are three small but significant changes to the [LASSERRE-VKOMPELLA] VPLS draft which would make the Service Provider Backbone much more compatible with a bridge-based PE implementation, and which would improve the efficiency of all L2VPN implementations.

Table of Contents

1. Introduction

A number of Ethernet Service Providers are currently building their networks using purely L2 technologies, based around bridges. When such an L2-oriented network provider looks at the architecture of [ANDERSSON], the similarity of Provider Edges (PEs) and bridges is unavoidable, and the desirability of constructing a PE based on a bridge is attractive. A bridge-based PE allows the Provider to make use of the extensive capabilities of the current generation of bridges.

Trying to base a PE implementation on a bridge raises certain issues with the specification and implementation of L2VPNs which have not been addressed in the drafts, to date. Resolving these issues requires some minor changes to [LASSERRE-VKOMPELLA]. These are:

1. Two new "forget MAC addresses" L2VPN control packets are needed.

2. "Forget MAC addresses" L2VPN control packets should be sent when backup links become activated, not when links fail.

3. It must be possible to configure associations among L2VPN instances such that a group of L2VPNs share a single MAC address database in any given attached device, but different groups of L2VPNs use different databases.

This present document assumes the validity of the [ANDERSSON] and [LASSERRE-VKOMPELLA] drafts. We use the terminology of [ANDERSSON], borrowing from [LASSERRE-VKOMPELLA] and [SAJASSI] when needed.

Sections 2 and 3 explain the need to have forms of the [LASSERRE-VKOMPELLA] flush message that are compatible with the operation of bridges, and the necessary timing of sending those messages. Section 4 describes a requirement to allow some L2VPNs to share a common MAC address database.

2. Signaling the Need to Unlearn MAC Addresses

The "flush" message of [LASSERRE-VKOMPELLA] is inadequate to the needs of bridges either serving as PEs or as part of an Access Network [ANDERSSON] in a Provider Network. The two forms of the flush message are, "forget all MAC addresses in this list," and "forget all the MAC addresses you learned from me." There is no way for a bridge to generate an accurate list of MAC addresses for the first message, and no circumstances under which a bridge would issue the second message.

The following sub-sections describe the need for unlearning, how the two major spanning tree protocols handle the deliberate forgetting of MAC address information, and the bridge requirements for flush messages.

2.1. Requirement for Unlearning MAC Addresses

If bridges operate over Psuedo Wires (PWs) such that redundant PEs are provided to improve the availability of the Provider Network, then the possibility arises that changes in an Ethernet Service Provider's Access Network will require packets to take different paths through the Provider Backbone. An obvious example is shown in Figure 1:

```
  +-------------+ .1Q +----------+ PW1  +----------+   +-------------+
  + PE-bridge 1 +------+  PE 1   +-------+  PE 3   +---+ PE-bridge 2 |
  +-+---------+-+     +--------+-+      +-+--------+   +-----------+-+
    | Ether    \          PW2 | Backbone /                   Ether |
    |           \ .1Q +--------+-+      /                          |
  Customer       '----+  PE 3   +-----'                      Customer
  Station A           +----------+ PW3                       Station B
```

FIGURE 1: Unlearning MAC addresses in L2VPN due to L2 changes

In this diagram, we have five Provider bridges, PE-bridge 1 and 2, and PEs 1 through 3. The PWs of one (data) VLPLS are shown. Keep in mind, however that we are not using spanning tree to select among PWs; the PWs form a full mesh, and split horizon is used to prevent loops within the L2VPN.

Spanning tree may, however, be running over the whole network, in which case spanning tree Bridge Protocol Data Units (BPDUs) may be carried as ordinary multicast data over one or more of the PWs.

Suppose that the normal path between stations A and B goes through PE 1. PE 3 has learned this fact, and directs its packets destined for station A to PE 1. If the link between PE-bridge 1 and PE 1 fails, then it is possible that the redundancy/failover algorithm in use will select PE 2, rather than PE 1, to be the portal to the Backbone. In that case, PE 3 needs to "unlearn" its association between MAC address A and PE 1.

2.2. How Bridges Forget MAC Addresses

The spanning tree algorithms in [802.1D] and [802.1w] provide two methods for notifying bridges when MAC address information needs to be unlearned. The "classic" Spanning Tree Protocol, STP, defined in [802.1D] describe one way, and the new Rapid Spanning Tree Protocol (RSTP) in [802.1w] behaves another way.

In STP, when a topology change occurs, notification of the change is transmitted to the root bridge, which in turn relays the fact to all bridges. The notification is not directional; the root places all bridges in "topology change mode" for a certain length of time, then returns all bridges to normal mode. While in topology change mode, all MAC address information is timed out over a much shorter period than is normally the case.

In normal times, the default timeout period for MAC address information is five minutes. During a topology change, that time shortens to a default value of 15 seconds. This time is comparable to the time during which service may be interrupted by a topology change. The shortened timeout period ensures that stale directionality information will not survive the topology change. Note that, if the MAC addresses were instantly forgotten, instead of timed out rapidly, a great deal of traffic otherwise unaffected by the topology change would be unnecessarily flooded.

In RSTP, a Topology Change Notification (TCN) is initiated only by a bridge port that has just transitioned from standby to operational status. It is generated only on that newly operational port, and is then relayed along the spanning tree by the other bridges. Thus, RSTP TCNs have a direction of propagation, and bridges "behind" the new link do not receive it. Since RSTP can converge in milliseconds after a topology change, receipt of a TCN causes a bridge to instantly forget many MAC addresses. A bridge does not forget MAC addresses associated with the port on which the TCN was received; one can prove that the MAC address information on

that port cannot be affected by any topology change.

2.3. Improved L2VPN Flush Messages

To be consistent with bridges, the L2VPN learning and forgetting rules must be compatible with the bridge rules described in the previous section. The two specific L2VPN messages required for full compatibility with all standard bridges are:

v.IP 1. STP TOPOLOGY CHANGE START/END. Set the timeout period of all learned MAC addresses on this list of L2VPNs to this number of seconds. This message is transmitted with a shortened timeout value at the beginning of a "classic" STP topology change event, and transmitted with the default timeout period after the event ends.

2. RSTP TOPOLOGY CHANGE. Immediately delete all MAC address information on this list of L2VPNs except that information learned from the sender of this message. (Note that this is exactly the opposite from the current "flush all you learned from the sender" message.)

The first message, which is compatible with the old STP, is of questionable utility, simply because it is needed by a form of STP which takes 10s of seconds to converge after the failure or recovery of a node or link. The Rapid Spanning tree converges much more quickly, and uses the second form.

To the best of the authors' knowledge, these two actions are sufficient to meet the needs of all proprietary Layer 2 failure/recovery protocols based on spanning tree. If not, suggestions for additional actions are encouraged.

3. Send Flush on Recovery, Not Failure

In [LASSERRE-VKOMPELLA], a device that notices the failure of a non-PW link is required to transmit the flush message(s) over the Backbone. It is much better to transmit the flush messages at recovery time, that is, when an alternate to the failed link becomes operational, or when a new link is added. In brief, one reason is that frames are best flooded when there is a good chance that their destinations are reachable, and therefore will elicit the replies that will terminate the flooding. The second reason is that link failures cannot reliably be detected, whereas recovery events are sure and certain.

In RSTP [802.1w], it is the device that starts using the backup link that generates the flush message(s).

```
         "backdoor" _________________________________
            link   /                                 \
               +--+-+     +----+  Y--+----+        +--+-+
               | B1 |-----| B2 |-----| B3 |........| B4 +--Z
               +----+  ___+--+-+     +--+-+,..   ..+----+
                     \/      |   ___/   |   : \ /    : (Backbone)
                     /\___   |  /       |   :  :     :
               +----+     +--+-+     +--+-+;../ \..+----+
           X --+ B5 |-----| B6 |-----| B7 |........| B8 |
               +----+     +----+     +----+        +----+
```

Figure 2: L2VPN Flush Messages

In Figure 2, consider a PE B3 with some form of Ethernet connections on one side, and the Pseudowire world on the other side. If that PE discovers that a link has gone down, say the B5-B2 link, the B2-B3 link, or the Y-B3 link, there are several possibilities for what can be wrong or right about learned MAC address information:

1.   The link is permanently down. Those MAC addresses will be unreachable for a very long time.

2.   The link is momentarily down. Those MAC addresses will again be reachable through the same link in a very short time.

3.   The link is down, but a backup link to this same PE, carrying those same MAC addresses, will very shortly be activated.

4   The link is down, but a backup link to another PE, carrying those same MAC addresses, will very shortly be activated.

In all of these cases, forgetting the MAC addresses immediately upon failure of the link does not help anything, especially if there are a large number of them associated with the failed link. If the MAC addresses "never" come back, they will eventually time out and be deleted everywhere. Assuming that the link was running at full speed when it failed, all of the traffic already in, queued for transmission to, or about to be sent by the user to the network, will be flooded throughout the L2VPN. Furthermore, this burst of flooding is useless, as it will occur at the very moment when it is the least likely that the flooded frames can reach their destinations. This means, of course, that the out-of-touch stations cannot respond to the flooded frames and put a stop the flooding. The flooding continues until the upper layers decide to wait for responses.

In fact, it is when a link transitions from backup to operational

status that MAC addresses can be profitably forgotten. Unless or until an alternate path to the lost MAC addresses becomes available, the packets destined for those MAC addresses is best black-holed. If the MAC addresses never come back, then in the usual case, the bridge MAC address timeouts are such that the upper layers will give up on the conversation before the bridges forget the MAC addresses and start flooding the doomed traffic. If the MAC addresses do come back, then old information is deleted as the first thing, and the flooded frames have a good chance of reaching their (new) destinations. This is why RSTP waits until a link comes up to generate a Topology Change Notification. Only when an alternate path is available is there any reason to flood frames everywhere.

Looking at Figure 2, suppose that links B5-B2, B2-B3, and the Backbone PWs are the primary links between B5 and B4, and specifically that B5-B6 is kept in reserve. In RSTP parlance, B5's port to B2 is a (forwarding) root port, and B5's port to B6 is a (discarding) alternate port. If the B5-B2 link fails, B5's port to B6 becomes the forwarding root port, and a TCN is sent. Note that, because of the direction of propagation of the TCN, B4 does not forget address Y, because that address cannot possibly have changed; the new link came up "behind" the Y-B3 connection. Interestingly, B3 does have to forget address Z, because it cannot know for certain that the "backdoor" link shown in the diagram has not come into use to deliver frames from Z via B1 and B2. In other words, RSTP does not assume that it knows the global topology. However, any such needlessly flooded frames will reach their destinations and be answered, and so will very quickly be relearned.

Many bridges treat MAC addresses attached to configured local access ports, rather than inter-bridge links, as sure knowledge. Those MAC addresses are not deleted from the owning bridge.

If the device that brings up the new link knows what MAC addresses it is serving, perhaps by configuration, then the best solution is to transmit a "learn the following MAC addresses" control message. This is ideal, and is provided by [LASSERRE-VKOMPELLA]. In general, however, a bridge does not know this.

One should also consider what happens if the failure occurs in a link not directly connected to the PE? What happens if the PE is connected to a shared medium Ethernet, so that the loss of another device's connection is invisible to the PE? (Shared media still exist, new ones such as packet rings are being created, and wireless hubs are very similar in nature.) What happens if only one side of the connection gets a "glitch" and believes that the

link has momentarily gone down?

The RSTP solution is known to handle all of these cases correctly. It would be better for any L2VPN solution to employ that same technique. In other words:

1. As "classic" STP enters and leaves the topology change mode, the bridge responsible for transmitting that fact to the Backbone should also transmit the "accelerate timeouts" L2VPN control message for all affected L2VPNs. {This is not needed if classic STP is not to be supported.)

2. When a "rapid" RSTP bridge transmits a TCN BPDU over the Backbone, it should also transmit the "forget all MAC addresses not learned from me" L2VPN control message for the Pseudowires associated with the affected spanning tree instance.

3. When some L2 device not utilizing spanning trees, such as the PE-CLE of [SAJASSI], switches to a new PE, it should transmit a control message towards the new PE. This message causes the PE to issue a "forget all MAC addresses not learned from me" L2VPN control message only for the L2VPNs connected to the affected PE-CLE.

4. Independent vs. Shared Address Learning

In order to be consistent with the current capabilities of [802.1Q] bridges, it must be possible to configure any number of L2VPNs either to use separate MAC address databases, as specified in [LASSERRE-VKOMPELLA], or to use the same MAC address database. If not, we remove a standard bridge capability that is not only expected by a significant fraction of the user community, but one which is actually more useful in the Ethernet Provider space than in the enterprise space.

In particular, a bridge which is conformant to [802.1Q] can be configured, for any given pair of VLANs, to store those two VLANs' MAC address information in two different Filtering Databases (Independent VLAN Learning, IVL), or to use the same Filtering Database for both (Shared VLAN Learning, SVL). (It also permits the user to specify "don't care", and let the bridge decide.) That is, MAC addresses learned on one VLAN may or may not, according to the specific configuration of the bridge, be used to forward frames on another VLAN. This fact has been utilized in a great many ways, by a number of bridge vendors and bridge customers, in order to implement a number of useful features. It is a behavior that has

been in IEEE 802.1Q since 1998, and in various vendors' bridges long before that date. It cannot be removed without a serious impact on the users of bridged LANs.

Given that one VLAN in the Access Network is associated with one L2VPN, we are lead to the conclusion that two or more L2VPNs may be similarly configured to use either IVL or SVL. In other words, two L2VPNs A and B may be configured such that a {MAC address, PW) association learned by a PE on L2VPN A is used when that PE transmits packets to a PW on L2VPN B. Another way to phrase this is that bridges do not remember {MAC, VLAN-ID, port} triplets, but instead, remember {MAC, FID, port} triplets, where "FID" is the Filtering ID, identifying a Filtering Database. The FID is derived from VLAN-ID through a statically configured mapping table. Similarly, L2VPN interfaces must learn {MAC, FID, PW} triplets instead of {MAC, L2VPN-ID, PW} triplets, by means of a configured L2VPN-to-Filtering-Database-ID table.

A concrete example of Shared VLAN Learning is the "Spanning Tree Per Bridge" technique which is most useful in a ring of bridges. (This is a more common topology in Ethernet MAN Provider Networks than in enterprise networks.) In a ring with one spanning tree, one link must be blocked, in order to prevent a closed forwarding loop. Traffic between bridges on opposite sides of the blocked link must go the long way around the ring. To avoid this inefficiency, the "Spanning Tree Per Bridge" technique employs n spanning trees, and splits each VLAN into a group of n VLANs, one for each spanning tree. Each bridge associates itself with exactly one spanning tree, selected so that the blocked link of that spanning tree is near the opposite side of the ring. Every frame it sends for a given VLAN group is sent only on the particular VLAN associated with that bridge's spanning tree. Clearly, for this to work, all of the VLANs in one group must share the same Filtering Database.

If the user of this "Spanning Tree Per Bridge" technique is a customer of an Ethernet MAN Service Provider, purchasing multiple L2VPNs to make the connections between the customer's bridges, then the provider's L2VPNs must share their learned MAC addresses.

Acknowledgements

The authors wish to thank Ali Sajassi, Joel Halpern, Steve Phillips and Adam Sweeney for their valuable suggestions, both technical and editorial, for correcting and improving this document, as well as a number of IEEE P802.1 voting members who reviewed it.

References

[ANDERSSON]
"PPVPN L2 Framework", draft-andersson-ppvpn-l2-framework-00.txt (Work in Progress)

[LASSERRE-VKOMPELLA]
"Virtual Private LAN Services over MPLS", draft-lasserre-vkompella-ppvpn-vpls-01.txt (Work in Progress)

[SAJASSI]
"VPLS Architectures", draft-sajassi-vpls-architectures-00.txt (Work in Progress)

[802.1D]
"Information technology. Telecommunications and information exchange between systems. Local and metropolitan area networks. Common specifications. Part 3: Media Access Control (MAC) Bridges", ANSI/IEEE Std 802.1D-1998.

[802.1Q]
"IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks", IEEE Std 802.1Q-1998.

[802.1w]
"IEEE Standard for Local and metropolitan area networks. Common specifications Part 3: Media Access Control (MAC) Bridges. Amendment 2: Rapid Reconfiguration", IEEE Std 802.1w-2001.

Authors' Addresses

Norman Finn
Cisco Systems
170 W Tasman Drive
San Jose, CA 95134
USA

Phone: +1.408.526.4495
Email: nfinn@cisco.com

Mick Seaman
Consultant
160 Bella Vista Ave
Belvedere
CA 94920

mick_seaman@ieee.org

Andrew Smith
Consultant

Email: ah_smith@acm.org
Fax: +1.415.345.1827

Allyn Romanow
Cisco Systems
170 W Tasman Drive
San Jose, CA 95134
USA

Phone +1.408.525.8836
Email: allyn@cisco.com

Full Copyright Statement