Network Working Group
Request for Comments: 5556
Category: Informational

J. Touch
USC/ISI
R. Perlman
Sun

May 2009

Transparent Interconnection of Lots of Links (TRILL): Problem and Applicability Statement

Status of This Memo

Copyright Notice

Abstract

Current IEEE 802.1 LANs use spanning tree protocols that have a number of challenges. These protocols need to strictly avoid loops, even temporary ones, during route propagation, because of the lack of header loop detection support. Routing tends not to take full advantage of alternate paths, or even non-overlapping pairwise paths (in the case of spanning trees). This document addresses these concerns and suggests applying modern network-layer routing protocols at the link layer. This document assumes that solutions would not address issues of scalability beyond that of existing IEEE 802.1 bridged links, but that a solution would be backward compatible with 802.1, including hubs, bridges, and their existing plug-and-play capabilities.

Table of Contents

1. Introduction

Conventional Ethernet networks -- known in the Internet as Ethernet link subnets -- have a number of attractive features, allowing hosts and routers to relocate within the subnet without requiring renumbering, and supporting automatic configuration. The basis of the simplicity of these subnets is the spanning tree, which although simple and elegant, can have substantial limitations. With spanning trees, the bandwidth across the subnet is limited because traffic flows over a subset of links forming a single tree -- or, with the latest version of the protocol and significant additional configuration, over a small number of superimposed trees. The oldest version of the spanning tree protocol can converge slowly when there are frequent topology changes.

The alternative to an Ethernet link subnet is often a network subnet. Network subnets can use link-state routing protocols that allow traffic to traverse least-cost paths rather than being aggregated on a spanning tree backbone, providing higher aggregate capacity and more resistance to link failures. Unfortunately, IP -- the dominant network layer technology -- requires that hosts be renumbered when relocated in different network subnets, interrupting network (e.g., tunnels, IPsec) and transport (e.g., TCP, UDP) associations that are in progress during the transition.

It is thus useful to consider a new approach that combines the features of these two existing solutions, hopefully retaining the desirable properties of each. Such an approach would develop a new kind of bridge system that was capable of using network-style routing, while still providing Ethernet service. It allows reuse of well-understood network routing protocols to benefit the link layer.

This document describes the challenge of such a combined approach. This problem is known as "Transparent Interconnection of Lots of Links" or "TRILL". The remainder of this document makes minimal assumptions about a solution to TRILL.

2. The TRILL Problem

Ethernet subnets have evolved from 'thicknet' to 'thinnet' to twisted pair with hubs to twisted pair with switches, becoming increasingly simple to wire and manage. Each level has corresponding topology restrictions; thicknet is inherently linear, whereas thinnet and hub-connected twisted pair have to be wired as a tree. Switches, added in IEEE 802.1D, allow network managers to avoid thinking in trees, where the spanning tree protocol finds a valid tree automatically; unfortunately, this additional simplicity comes with a number of associated penalties [Pe99].

The spanning tree often results in inefficient use of the link topology; traffic is concentrated on the spanning tree path, and all traffic follows that path even when other more direct paths are available. The addition in IEEE 802.1Q of support for multiple spanning trees helps a little, but the use of multiple spanning trees requires additional configuration, the number of trees is limited, and these defects apply within each tree regardless. The spanning tree protocol reacts to certain small topology changes with large effects on the reconfiguration of links in use. Each of these aspects of the spanning tree protocol can cause problems for current link-layer deployments.

2.1. Inefficient Paths

The Spanning Tree Protocol (STP) helps break cycles in a set of interconnected bridges, but it also can limit the bandwidth among that set and cause traffic to take circuitous paths. For example, in a set of N nodes that are interconnected pairwise along a ring, a spanning tree will disable one physical link so that connectivity is loop free. This will cause traffic between the pair of nodes connected by that disabled link to have to go N-1 physical hops around the entire remainder of the ring rather than take the most efficient single-hop path. Using modern routing protocols with such a topology, no traffic should have to go more than N/2 hops.

For another example, consider the network shown in Figure 1, which shows a number of bridges and their interconnecting links. End-hosts and routers are not shown; they would connect to the bridges that are shown, labeled A-H. Note that the network shown has cycles that would cause packet storms if hubs (repeaters) were used instead of spanning-tree-capable bridges. One possible spanning tree is shown by double lines.

```
                              [A]
                             // \    [C]
                            //   \   / \\  [D]
                           //     \ /   \\ //
                          [B]=====[H]=====[E]
                            \     //      ||
                             \   //       ||
                              \ //        ||
                               [G]--------[F]
```

Figure 1: Bridged subnet with spanning tree shown

The spanning tree limits the capacity of the resulting subnet. Assume that the links are 100 Mbps. Figure 2 shows how traffic from hosts on A to hosts on C goes via the spanning tree path A-B-H-E-C (links replaced with '1' in the figure); traffic from hosts on G to F go via the spanning three path G-H-E-F (links replaced by '2' in the figure). The link H-E is shared by both paths (alternating '1's and '2's), resulting in an aggregate capacity for both A..C and G..F paths of a total of 100 Mbps.

```
                                  [A]
                                  1           [C]
                                 1              1
                                1                1
                              [B]1111111[H]121212[E]
                                     2       2
                                    2        2
                                   2         2
                                  [G]       [F]
```

Figure 2: Traffic from A..C (1) and G..F (2) share a link

If traffic from G to F were to go directly using full routing, e.g., from G-F, both paths could have 100 Mbps each, and the total aggregate capacity could be 200 Mbps (Figure 3). In this case, the

H-F link carries only A-C traffic ('1's) and the G-F traffic ('2's) is more direct.

```
                                  [A]
                                  1           [C]
                                 1              1
                                1                1
                              [B]1111111[H]111111[E]



                                  [G]2222222[F]
```

Figure 3: Traffic from A..C (1) and G..F (2) with full routing

There are a number of features of modern layer 3 routing protocols which would be beneficial if available at layer 2, but which cannot practically be integrated into the spanning tree system such as multipath routing discussed in Section 2.2 below. Layer 3 routing typically optimizes paths between pairs of endpoints based on a cost metric, conventionally based on bandwidth, hop count, latency, and/or policy measures.

2.2. Multipath Forwarding

The discussion above assumes that all traffic flowing from one point to another follows a single path. Using spanning trees reduces aggregate bandwidth by forcing all such paths onto one tree, while modern routing causes such paths to be selected based on a cost metric. However, extensions to modern routing protocols enable even greater aggregate bandwidth by permitting traffic flowing from one endpoint to another to be sent over multiple, typically equal-cost, paths. (Traffic sent over different paths will generally encounter different delays and may be reordered with respect to traffic on another path. Thus, traffic must be divided into flows, such that reordering of traffic between flows is not significant, and those flows are allocated to paths.)

Multipathing typically spreads the traffic more evenly over the available physical links. The addition of multipathing to a routed network would typically result in only a small improvement in capacity for a network with roughly equal traffic between all pairs of nodes, because in that situation traffic is already fairly well dispersed. Conversely, multipathing can produce a dramatic improvement in a routed network where the traffic between a small number of pairs of nodes dominates, because such traffic can -- under the right circumstances -- be spread over multiple paths that might otherwise be lightly loaded.

2.3. Convergence and Safety

The spanning tree is dependent on the way a set of bridges are interconnected, i.e., the link-layer topology. Small changes in this topology can cause large changes in the spanning tree. Changes in the spanning tree can take time to propagate and converge, especially for older versions of STP.

One possible case occurs when one of the branches connected to the root bridge fails, causing a large number of ports to block and unblock before the network reconverges [Me04]. Consider a ring with a stub as shown in Figure 4.

```
                 [R]----[A]----[B]----[C]----[D]----[E]
                         |                           |
                         +--------[F]-----[G]--------+
```

Figure 4: Ring with poor convergence under reconfiguration

If A is the root bridge, then the paths A->B->C->D and A->F->G->E are the two open paths, while the D->E link is blocked. If the A->B link fails, then E must unblock its port to D for traffic to flow again, but it may require recomputation of the entire tree through BPDUs (Bridge PDUs). Even worse, if R is root and R or the A-R connection fails, BPDU updates related to the old and new root can lead to a brief count-to-infinity event, and, if RSTP (Rapid Spanning Tree Protocol) is in use, can delay convergence for a few seconds. The original IEEE 802.1 spanning tree protocol can impose 30-second delays in re-establishing data connectivity after a topology change in order to be sure a new topology has stabilized and been fully propagated.

The spanning tree protocol is inherently global to an entire layer 2 subnet; there is no current way to contain, partition, or otherwise factor the protocol into a number of smaller, more stable subsets that interact as groups. Contrast this with Internet routing, which includes both intradomain and interdomain variants, split to provide exactly that containment and scalability within a domain while allowing domains to interact freely independent of what happens within a domain.

2.4. Stability of IP Multicast Optimization

Although it is a layer violation, it is common for high-end bridges to snoop on IP multicast control messages for the purpose of optimizing the distribution of IP multicast data and of those control messages [RFC4541].

When such snooping and optimization is performed by spanning-tree-based bridges, it done at each bridge based on the traffic observed on that bridge’s ports. Changes in topology may reverse or otherwise change the required forwarding ports of messages for a multicast group. Bridges must relearn the correct multicast forwarding from the receipt of multicast control messages on new ports. Such control messages are sent to establish multicast distribution state and then to refresh it, sometimes at intervals of seconds. If a bridging topology change has occurred during such intervals, multicast data may be misdirected and lost.

However, a solution based on link-state routing, for example, can form and maintain a global view of the multicast group membership and multicast router situation in a similar fashion to that in which it maintains a global view of the status of links. Thus, such a solution can adjust the forwarding of multicast data and control traffic immediately as it sees the LAN topology change.

2.5. IEEE 802.1 Bridging Protocols

There have been a variety of IEEE protocols beyond the initial shared-media Ethernet variant, including:

- 802.1D - added bridges (i.e., switches) and a spanning tree protocol (STP) (incorporates 802.1w, below) [IEEE04].

- 802.1w - extension for rapid reconvergence of the spanning tree protocol (RTSP) [IEEE04].

- 802.1Q - added VLAN and priority support, where each link address maps to one VLAN (incorporates 802.1v and 802.1s, below) [IEEE06].

- 802.1v - added VLANs where segments map to VLANs based on link address together with network protocol and transport port [IEEE06].

- 802.1s - added support for multiple spanning trees, up to a maximum of 65, one per non-overlapping group of VLANs (Multiple STP) [IEEE06].

This document presumes the above variants are supported on the Ethernet subnet, i.e., that a TRILL solution would not interfere with (i.e., would not affect) any of the above.

In addition, the following more recent extensions have been standardized to specify provider/carrier Ethernet services that can be effectively transparent to the previously specified customer Ethernet services. The TRILL problem as described in this document is limited to customer Ethernet services; however, there is no reason that a TRILL solution might not be easily applicable to both customer and provider Ethernet.

- 802.1ad (Provider Bridges) - added support for a second level of VLAN tag, called a "service tag", and renamed the original 802.1Q tag a "customer tag". Also known as Q-in-Q because of the stacking of 802.1Q VLAN tags.

- 802.1ah (Provider Backbone Bridges) - added support for stacking of MAC addresses by providing a tag to contain the original source and destination MAC addresses. Also know as MAC-in-MAC.

It is useful to note that no extension listed above in this section addresses the issue of independent, localized routing in a single LAN -- which is the focus of TRILL.

The TRILL problem and a sketch of a possible solution [Pe04] were presented to both the IETF (via a BoF) and IEEE 802 (via an IEEE 802 Plenary Meeting Tutorial). The IEEE, in response, approved a project called Shortest Path Bridging (IEEE Project P802.1aq), taking a different approach than that presented in [Pe04]. The current Draft of P802.1aq appears to describe two different techniques. One, which does not use encapsulation, is, according to the IEEE Draft, limited in applicability to small networks of no more than 100 shortest path bridges. The other, which uses 802.1ah, is, according to the IEEE Draft, limited in applicability to networks of no more than 1,000 shortest path bridges.

2.6. Problems Not Addressed

There are other challenges to deploying Ethernet subnets that are not addressed in this document other than, in some cases, to mention relevant IEEE 802.1 documents, although it is possible for a solution to address one or more of these in addition to the TRILL problem. These include:

- increased Ethernet link subnet scale

- increased node relocation

- security of the Ethernet link subnet management protocol

- flooding attacks on a Ethernet link subnet

- support for "provider" services such as Provider Bridges (802.1ad), Provider Backbone Bridges (802.1ah), or Provider Backbone Bridge Traffic Engineering (802.1Qay)

Solutions to TRILL need not support deployment of larger scales of Ethernet link subnets than current broadcast domains can support (e.g., around 1,000 end-hosts in a single bridged LAN of 100 bridges, or 100,000 end-hosts inside 1,000 VLANs served by 10,000 bridges).

Similarly, solutions to TRILL need not address link-layer node migration, which can complicate the caches in learning bridges. Similar challenges exist in the Address Resolution Protocol (ARP), where link-layer forwarding is not updated appropriately when nodes move to ports on other bridges. Again, the compartmentalization available in network routing, like that of network-layer Autonomous Systems (ASes), can help hide the effect of migration. That is a side effect, however, and not a primary focus of this work.

Current link-layer control-plane protocols, including Ethernet link subnet management (spanning tree) and link/network integration (ARP), are vulnerable to a variety of attacks. Solutions to TRILL need not address these insecurities. Similar attacks exist in the data plane, e.g., source address spoofing, single address traffic attacks, traffic snooping, and broadcast flooding. TRILL solutions need not address any of these issues, although it is critical that they do not introduce new vulnerabilities in the process (see Section 5).

3. Desired Properties of Solutions to TRILL

This section describes some of the desirable or required properties of any system that would solve the TRILL problems, independent of the details of such a solution. Most of these are based on retaining useful properties of bridges, or maintaining those properties while solving the problems listed in Section 2.

3.1. No Change to Link Capabilities

There must be no change to the service that Ethernet subnets already provide as a result of deploying a TRILL solution. Ethernet supports unicast, broadcast, and multicast natively. Although network protocols, notably IP, can tolerate link layers that do not provide all three, it would be useful to retain the support already in place [RFC3819]. So called "zero configuration protocols" (also known as "zeroconf", e.g., as used to configure link-local addresses [RFC3927]), as well as existing bridge autoconfiguration, are also dependent on broadcast.

Current Ethernet ensures in-order delivery for frames of the same priority and no duplicated frames, under normal operation (excepting transients during reconfiguration). These criteria apply in varying degrees to the different types of Ethernet, e.g., basic Ethernet up through basic VLAN (802.1Q) ensures that all frames with the same priority between two link addresses have both properties, but protocol/port VLAN (802.1v) ensures this only for packets with the same protocol and port. There are subtle implications to such a requirement. Bridge autolearning already is susceptible to moving nodes between ports, because previously learned associations between the port and link address change. A TRILL solution could be similarly susceptible to such changes.

3.2. Zero Configuration and Zero Assumption

Both bridges and hubs are zero configuration devices; hubs having no configuration at all, and bridges being automatically self-configured. Bridges are further zero-assumption devices, unlike hubs. Bridges can be interconnected in arbitrary topologies, without regard for cycles or even self-attachment. Spanning tree protocols (STPs) remove the impact of cycles automatically, and port autolearning reduces unnecessary broadcast of unicast traffic.

A TRILL solution should strive to have a similar zero-configuration, zero-assumption operation. This includes having TRILL solution components automatically discover other TRILL solution components and organize themselves, as well as to configure that organization for proper operation (plug-and-play). It also includes zero-configuration backward compatibility with existing bridges and hubs, which may include interacting with some of the bridge protocols, such as spanning tree.

VLANs add a caveat to zero configuration; a TRILL solution should support automatic use of a default VLAN (like non-VLAN bridges), but would undoubtedly require explicit configuration for VLANs where bridges require such configuration.

Autoconfiguration extends to optional services, such as multicast support via Internet Group Management Protocol (IGMP) snooping, broadcast support via serial copy, and support of multiple VLANs.

3.3. Forwarding Loop Mitigation

Using spanning trees avoids forwarding loops by construction, although transient loops can occur, e.g., via the temporarily undetected appearance of new link connectivity or the loss of a sufficient number of spanning-tree control frames. Solutions to TRILL are intended to use adapted network-layer routing protocols that may introduce transient loops during routing convergence. A TRILL solution thus needs to provide support for mitigating the effect of such routing loops.

In the Internet, loop mitigation is provided by decrementing hop counts (Time To Live (TTL)); in other networks, packets include a trace (sometimes referred to as ’serialized’ or ’unioned’) of visited nodes [RFC1812]. In addition, there may be localized consistency checks, such as whether traffic is received on an unexpected interface, which indicates that routing is in flux and that such traffic should probably be discarded for safety. These types of mechanisms limit the impact of loops or detect them explicitly. Mechanisms with similar effect should be included in TRILL solutions.

3.4. Spanning Tree Management

In order to address convergence under reconfiguration and robustness to link interruption (Section 2.2), participation in the spanning tree (STP) must be carefully managed. The goal is to provide the desired stability of the TRILL solution and of the entire Ethernet link subnet, which may include bridges using STP. This may involve a TRILL solution participating in the STP, where the protocol used for TRILL might dampen interactions with STP, or it may involve severing the STP into separate STPs on ’stub’ external Ethernet link subnet segments.

A requirement is that a TRILL solution must not require modifications or exceptions to the existing spanning tree protocols (e.g., STP, RSTP (Rapid Spanning Tree Protocol), MSTP (Multiple Spanning Tree Protocol)).

3.5. Multiple Attachments

In STP, a single node with multiple attachments to a single spanning tree segment will always get and send traffic over only one of the those attachment points. TRILL must manage all traffic, including multicast and broadcast traffic, so as not to create traffic loops involving Ethernet segments with multiple TRILL attachment points. This includes multiple attachments to a single TRILL node and attachments to multiple TRILL nodes. Support for multiple attachments can improve support for forms of mobility that induce topology changes, such as "make before break", although this is not a major goal of TRILL.

3.6. VLAN Issues

A TRILL solution should support multiple customer VLANs (802.1Q, which includes 802.1v and 802.1s). This may involve ignorance, just as many bridge devices do not participate in the VLAN protocols. A TRILL solution may alternately furnish direct VLAN support, e.g., by providing configurable support for VLAN-ignorant end stations equivalent to that provided by 802.1Q non-provider bridges.

Provider VLANs (802.1ad) are outside of the scope of this document. A TRILL solution might or might not be easily adaptable to handling provider VLANs.

3.7. Operational Equivalence

As with any extension to an existing architecture, it would be useful -- though not strictly necessary -- to be able to describe or consider a TRILL solution as equivalent to an existing link layer component. Such equivalence provides a validation model for the architecture and a way for users to predict the effect of the use of a TRILL solution on a deployed Ethernet. In this case, 'user' refers to users of the Ethernet protocol, whether at the host (data segments), bridge (ST control segments), or VLAN (VLAN control).

This provides a sanity check, i.e., "we got it right if we can exchange a TRILL solution component or components with an X" (where "X" might be a single bridge, a hub, or some other link layer abstraction). It does not matter whether "X" can be implemented on the same scale as the corresponding TRILL solution. It also does not matter if it can -- there may be utility to deploying the TRILL solution components incrementally, in ways that a single "X" could not be installed.

For example, if a TRILL solution's components were equivalent to a single IEEE 802.1D bridge, it would mean that they would -- as a whole - participate in the STP. This need not require that TRILL solution components would propagate STP, any more than a bridge need do so in its on-board control. It would mean that the solution would interact with BPDUs at the edge, where the solution would -- again, as a whole - participate as if a single node in the spanning tree. Note that this equivalence is not required; a solution may act as if an IEEE 802.1 hub, or may not have a corresponding equivalent link layer component at all.

3.8. Optimizations

There are a number of optimizations that may be applied to TRILL solutions. These must be applied in a way that does not affect functionality as a tradeoff for increased performance. Such optimizations may address broadcast and multicast frame distribution, VLAN support, and snooping of ARP and IPv6 neighbor discovery.

In addition, there may be optimizations which make the implementation of a TRILL solution easier than roughly equivalent existing bridge devices. For example, in many bridged LANs, there are topologies such that central ("core") bridges which have both a greater volume of traffic flowing through them as well as traffic to and from a larger variety of end station than do non-core bridges. Thus means that such core bridges need to learn a large number of end station addresses and need to do lookups based on such addresses very rapidly. This might require large high speed content addressable memory making implementation of such core bridges difficult. Although a TRILL solution need not provide such optimizations, it may reduce the need for such large, high speed content addressable memories or provide other similar optimizations.

3.9. Internet Architecture Issues

TRILL solutions are intended to have no impact on the Internet network layer architecture. In particular, the Internet and higher layer headers should remain intact when traversing a deployed TRILL solution, just as they do when traversing any other link subnet technologies. This means that the IP TTL field cannot be co-opted for forwarding loop mitigation, as it would interfere with the Internet layer assuming that the link subnet was reachable with no changes in TTL. (Internet TTLs are changed only at routers, as per RFC 1812, and even if IP TTL were considered, TRILL is expected to support non-IP payloads, and so requires a separate solution anyway [RFC1812]).

TRILL solutions should also have no impact on Internet routing or signaling, which also means that broadcast and multicast, both of which can pervade an entire Ethernet link subnet, must be able to transparently pervade a deployed TRILL solution. Changing how either of these capabilities behaves would have significant effects on a variety of protocols, including RIP (broadcast), RIPv2 (multicast), ARP (broadcast), IPv6 neighbor discovery (multicast), etc.

Note that snooping of network-layer packets may be useful, especially for certain optimizations. These include snooping multicast control-plane packets (IGMP) to tune link multicast to match the network multicast topology, as is already done in existing smart switches [RFC3376] [RFC4286]. This also includes snooping IPv6 neighbor discovery messages to assist with governing TRILL solution edge configuration, as is the case in some smart learning bridges [RFC4861]. Other layers may similarly be snooped, notably ARP packets, for similar reasons as for IPv4 [RFC826].

4. Applicability

As might be expected, TRILL solutions are intended to be used to solve the problems described in Section 2. However, not all such installations are appropriate environments for such solutions. This section outlines the issues in the appropriate use of these solutions.

TRILL solutions are intended to address problems of path efficiency and concentration, inability to multipath, and path stability within a single Ethernet link subnet. Like bridges, individual TRILL solution components may find other TRILL solution components within a single Ethernet link subnet and aggregate into a single TRILL solution.

TRILL solutions are not intended to span separate Ethernet link subnets interconnected by network-layer (e.g., router) devices, except via link-layer tunnels, where such tunnels render the distinct subnet undetectably equivalent from a single Ethernet link subnet.

A currently open question is whether a single Ethernet link subnet should contain components of only one TRILL solution, either of necessity of architecture or utility. Multiple TRILL solutions, like Internet ASes, may allow TRILL routing protocols to be partitioned in ways that help their stability, but this may come at the price of needing the TRILL solutions to participate more fully as nodes (each modeling a bridge) in the Ethernet link subnet STP. Each architecture solution should decide whether multiple TRILL solutions are supported within a single Ethernet link subnet, and mechanisms should be included to enforce whatever decision is made.

TRILL solutions need not address scalability limitations in bridged subnets. Although there may be scale benefits of other aspects of solving TRILL problems, e.g., of using network-layer routing to provide stability under link changes or intermittent outages, this is not a focus of this work.

As also noted earlier, TRILL solutions are not intended to address security vulnerabilities in either the data plane or control plane of the link layer. This means that TRILL solutions should not limit broadcast frames, ARP requests, or spanning tree protocol messages (if such are interpreted by the TRILL solution or solution edge).

5. Security Considerations

TRILL solutions should not introduce new vulnerabilities compared to traditional bridged subnets.

TRILL solutions are not intended to be a solution to Ethernet link subnet vulnerabilities, including spoofing, flooding, snooping, and attacks on the link control plane (STP, flooding the learning cache) and link-network control plane (ARP). Although TRILL solutions are intended to provide more stable routing than STP, this stability is limited to performance, and the subsequent robustness is intended to address non-malicious events.

There may be some side-effects to the use of TRILL solutions that can provide more robust operation under certain attacks, such as those interrupting or adding link service, but TRILL solutions should not be relied upon for such capabilities.

Finally, TRILL solutions should not interfere with other protocols intended to address these vulnerabilities, such as those to secure IPv6 neighbor discovery [RFC3971].

6. Acknowledgments

Portions of this document are based on documents that describe a preliminary solution, and on a related network-layer solution [Pe04] [Pe05] [To03]. Donald Eastlake III provided substantial text and comments. Additional comments and feedback were provided by the members of the IETF TRILL WG, in which this document was developed, and by the IESG.

This document was prepared using 2-Word-v2.0.template.dot.

7. Informative References

[IEEE04] IEEE 802.1D bridging standard, "IEEE Standard for Local and metropolitan area networks: Media Access Control (MAC) Bridges", (incorporates 802.1w), Jun. 2004.

[IEEE06] IEEE 802.1Q VLAN standard, "IEEE Standards for Local and metropolitan area networks: Virtual Bridged Local Area Networks", (incorporates 802.1v and 802.1s), May 2006.

[Me04] Myers, A., T.E. Ng, H. Zhang, "Rethinking the Service Model: Scaling Ethernet to a Million Nodes", Proc. ACM Third Workshop on Hot Topics in Networks (HotNets-III), Mar. 2004.

[Pe99] Perlman, R., "Interconnection: Bridges, Routers, Switches, and Internetworking Protocols", Addison Wesley, Chapter 3, 1999.

[Pe04] Perlman, R., "RBridges: Transparent Routing", Proc. Infocom 2005, Mar. 2004.

[Pe05] Perlman, R., J. Touch, A. Yegin, "RBridges: Transparent Routing," (expired work in progress), Apr. 2004 - May 2005.

[RFC826] Plummer, D., "Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware", STD 37, RFC 826, November 1982.

[RFC1812] Baker, F., Ed., "Requirements for IP Version 4 Routers", RFC 1812, June 1995.

[RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, October 2002.

[RFC3819] Karn, P., Ed., Bormann, C., Fairhurst, G., Grossman, D., Ludwig, R., Mahdavi, J., Montenegro, G., Touch, J., and L. Wood, "Advice for Internet Subnetwork Designers", BCP 89, RFC 3819, July 2004.

[RFC3927] Cheshire, S., Aboba, B., and E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses", RFC 3927, May 2005.

[RFC3971] Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC4286] Rosenberg, J., "Extensible Markup Language (XML) Formats for Representing Resource Lists", RFC 4826, May 2007.

[RFC4541] Christensen, M., Kimball, K., and F. Solensky, "Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches", RFC 4541, May 2006.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[To03] Touch, J., Y. Wang, L. Eggert, G. Finn, "A Virtual Internet Architecture", ISI Technical Report ISI-TR-570, Presented at the Workshop on Future Directions in Network Architecture (FDNA) 2003 at Sigcomm 2003, March 2003.

Authors' Addresses

Joe Touch
USC/ISI
4676 Admiralty Way
Marina del Rey, CA 90292-6695
U.S.A.

Phone: +1 (310) 448-9151
EMail: touch@isi.edu
URL: http://www.isi.edu/touch

Radia Perlman
Sun Microsystems
16 Network Circle
umpk16-161
Menlo Park, CA 94025
U.S.A.

EMail: Radia.Perlman@sun.com