PANRG T. Enghardt
Internet-Draft TU Berlin
Intended status: Informational C. Kraehenbuehl
Expires: April 21, 2019 ETH Zuerich
October 18, 2018

A Vocabulary of Path Properties
draft-enghardt-panrg-path-properties-00

Abstract

This document defines and categorizes information about Internet paths that an endpoint might have or want to have. This information is expressed as properties of paths between two endpoints.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Because the current Internet provides an IP-based best-effort bit pipe, endpoints have little information about paths to other endpoints. A Path Aware Network exposes information about one or multiple paths through the network to endpoints, so that endpoints can use this information.

Such path properties may be relatively dynamic, e.g. current Round Trip Time, close to the origin, e.g. nature of the access technology on the first hop, or far from the origin, e.g. list of ASes traversed.

Usefulness over time is fundamentally different for dynamic and non-dynamic properties. The merit of a momentary measurement of a dynamic path property diminishes greatly as time goes on, e.g. the merit of an RTT measurement from a few seconds ago is quite small, while a non-dynamic path property might stay relevant, e.g. a NAT can be assumed to stay on a path during the lifetime of a connection, as the removal of the NAT would break the connection.

Non-dynamic properties are further separated into (local) domain properties related to the first few hops of the connection, and backbone properties related to the remaining hops. Domain properties expose a high amount of information to endpoints and strongly influence the connection behavior while there is little influence and information about backbone properties.

Dynamic properties are not separated into domain and backbone properties, since most of these properties are defined for a complete path and it is difficult and seldom useful to define them on part of the path. There are exceptions such as dynamic wireless access properties, but these do not justify separation into different categories.

This document addresses the first of the questions in Path-Aware Networking [I-D.irtf-panrg-questions], which is a product of the PANRG in the IRTF.

2. Domain Properties

Domain path properties usually relate to the access network within the first hop or the first few hops. Endpoints can influence domain properties for example by switching from a WiFi to a cellular interface, changing their data plan to increase throughput, or moving closer to a wireless access point which increases the signal strength.

A large amount of information about domain properties exists. Properties related to configuration can be queried using provisioning domains (PvDs). A PvD is a consistent set of network configuration information as defined in [RFC7556], e.g., relating to a local network interface. This may include source IP address prefixes, IP addresses of DNS servers, name of an HTTP proxy server, DNS suffixes associated with the network, or default gateway IP address. As one PvD is not restricted to one local network interface, a PvD may also apply to multiple paths.

Access Technology present on the path: The lower layer technology on the first hop, for example, WiFi, Wired Ethernet, or Cellular. This can also be more detailed, e.g., further specifying the Cellular as 2G, 3G, 4G, or 5G technology, or the WiFi as 802.11a, b, g, n, or ac. These are just examples, this list is not exhaustive, and there is no common index of identifiers here. Note that access technologies further along the path may also be relevant, e.g., a cellular backbone is not only the first hop, and there may be a DSL line behind the WiFi.

Monetary Cost: This is information related to billing, data caps, etc. It could be the allowed monthly data cap, the start and end of a billing period, the monetary cost per Megabyte sent or received, etc.

3. Backbone Properties

Backbone path properties relate to non-dynamic path properties that are not within the endpoint's domain. They are likely to stay constant within the lifetime of a connection, since Internet "backbone" routes change infrequently. These properties usually change on the timescale of seconds, minutes, or hours, when the route changes.

Even if these properties change, endpoints can neither specify which backbone nodes to use, nor verify data was sent over these nodes. An endpoint can for example choose its access provider, but cannot choose the backbone path to a given destination since the access provider will make their own policy-based routing decision.

Presence of certain device on the path: Could be the presence of a certain kind of middlebox, e.g., a proxy, a firewall, a NAT.

Presence of a packet forwarding node or specific Autonomous System on a path:
Indicates that traffic goes through a certain node or AS, which might be relevant for deciding the level of trust this path provides.

Disjointness: How disjoint a path is from another path.

Path MTU: The end-to-end maximum transmission unit in one packet.

Transport Protocols available: Whether a specific transport protocol can be used to establish a connection over this path. An endpoint may know this because it has cached whether it could successfully establish, e.g., a QUIC connection, or an MPTCP subflow.

Protocol Features available: Whether a specific feature within a protocol is known to work over this path, e.g., ECN, or TCP Fast Open.

4. Dynamic Properties

Dynamic Path Properties are expected to change on the timescale of milliseconds. They usually relate to the state of the path, such as the currently available end-to-end bandwidth. Some of these properties may depend only on the first hop or on the access network, some may depend on the entire path.

Typically, Dynamic Properties can only be approximated and sampled, and might be made available in an aggregated form, such as averages or minimums. Dynamic Path Properties can be measured by the endpoint itself or somethere in the network. See [ANRW18-Metrics] for a discussion of how to measure some dynamic path properties at the endpoint.

These properties may be symmetric or asymmetric. For example, an asymmetric property may be different in the upstream direction and in the downstream direction from the point of view of a particular host.

Available bandwidth: Maximum number of bytes per second that can be sent or received over this path. This depends on the available bandwidth at the bottleneck, and on crosstraffic.

Round Trip Time: Time from sending a packet to receiving a response from the remote endpoint.

Round Trip Time variation: Disparity of Round Trip Time values either over time or among multiple concurrent connections. A high RTT variation often indicates congestion.

Packet Loss: Percentage of sent packets that are not received on the other end.

Congestion: Whether there is any indication of congestion on the path.

Wireless Signal strength: Power level of the wireless signal being received. Lower signal strength, relative to the same noise floor, is correlated with higher bit error rates and lower modulation rates.

Wireless Modulation Rate: Modulation bitrate of the wireless signal. The modulation rate determines how many bytes are transmitted within a symbol on the wireless channel. A high modulation rate leads to a higher possible bitrate, given sufficient signal strength.

Wireless Channel utilization: Percentage of time during which there is a transmission on the wireless medium. A high channel utilization indicates a congested wireless network.

5. Security Considerations

Some of these properties may have security implications for endpoints. For example, a corporate policy might require to have a firewall on the path.

For properties provided by the network, their authenticity and correctness may need to be verified by an endpoint.

6. IANA Considerations

This document has no IANA actions.

7. Informative References

[ANRW18-Metrics]
Enghardt, T., Tiesel, P., and A. Feldmann, "Metrics for access network selection", Proceedings of the Applied Networking Research Workshop on - ANRW '18, DOI 10.1145/3232755.3232764, 2018.

[I-D.irtf-panrg-questions]
Trammell, B., "Open Questions in Path Aware Networking", draft-irtf-panrg-questions-00 (work in progress), April 2018.

[RFC7556] Anipko, D., Ed., "Multiple Provisioning Domain Architecture", RFC 7556, DOI 10.17487/RFC7556, June 2015, <https://www.rfc-editor.org/info/rfc7556>.

Acknowledgments

Thanks to Paul Hoffman for feedback and editorial changes.

Authors' Addresses

Theresa Enghardt
TU Berlin

Email: theresa@inet.tu-berlin.de

Cyrill Kraehenbuehl
ETH Zuerich

Email: cyrill.kraehenbuehl@inf.ethz.ch

PANRG T. Enghardt
Internet-Draft TU Berlin
Intended status: Informational C. Kraehenbuehl
Expires: May 7, 2020 ETH Zuerich
November 04, 2019

A Vocabulary of Path Properties
draft-enghardt-panrg-path-properties-03

Abstract

Path properties express information about paths across a network and the services provided via such paths. In a path-aware network, path properties may be fully or partially available to entities such as hosts. This document defines and categorizes path properties. Furthermore, the document specifies several path properties which might be useful to hosts or other entities.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

In the current Internet architecture, hosts generally do not have information about forwarding paths through the network and about services associated with these paths. A path-aware network, as introduced in [I-D.irtf-panrg-questions], exposes information about paths to hosts or to other entities. This document defines such information as path properties, addressing the first of the questions in path-aware networking [I-D.irtf-panrg-questions].

As terms related to paths have different meanings in different areas of networking, first, this document provides a common terminology to define paths, path elements, and path properties. Then, this document provides some examples for use cases for path properties. Finally, the document lists several path properties that may be useful for the mentioned use cases.

2. Terminology

Node: An entity which processes packets, e.g., sends, receives, forwards, or modifies them. A node may be physical or virtual, e.g., a machine or a service function. A node may also be the collection of multiple entities which, as a collection, processes packets, e.g., an entire Autonomous System (AS).

Host: A node that generally executes application programs on behalf of user(s), employing network and/or Internet communication services in support of this function, as defined in [RFC1122].

Link: A medium or communication facility that connects two or more nodes with each other. A link enables a node to send packets to other nodes. Links can be physical, e.g., a WiFi network which connects an Access Point to stations, or virtual, e.g., a virtual switch which connects two virtual machines hosted on the same physical machine. A link is unidirectional and bidirectional communication can be modeled as two links between the same nodes in opposite directions.

Path element: Either a node or a link.

Path: A sequence of adjacent path elements over which a packet can be transmitted, starting and ending with a node. Paths are time-dependent, i.e., the sequence of path elements over which packets are sent from one node to another may change frequently. A path is defined between two nodes. For multicast or broadcast, a packet may be sent by one node and received by multiple nodes. In this case, the packet is sent over multiple paths at once, one path for each combination of sending and receiving node. Note that an entity may have only partial visibility of the path elements that comprise a path, and entities may treat path elements at different levels of abstraction.

Subpath: Given a path, a subpath is a sequence of adjacent path elements of this path.

Flow: An entity made of packets to which the traits of a path or set of subpaths may be applied in a functional sense. For example, a flow can consist of all packets sent within a TCP session with the same five-tuple between two hosts, or it can consist of all packets sent on the same physical link.

Property: A trait of one or a sequence of path elements, or a trait of a flow with respect to one or a sequence of path elements. An example of a link property is the maximum data rate that can be sent over the link. An example of a node property is the administrative domain that the node belongs to. An example of a property of a flow with respect to a subpath is the aggregated one-way delay of the flow being sent from one node to another node over this subpath. A property is thus described by a tuple containing the path element(s), the flow or an empty set if no packets are relevant for the property, the name of the property (e.g., maximum data rate), and the value of the property (e.g., 1Gbps).

Aggregated property: A collection of multiple values of a property into a single value, according to a function. A property can be aggregated over multiple path elements (i.e., a path), e.g., the

MTU of a path as the minimum MTU of all links on the path, over multiple packets (i.e., a flow), e.g., the median one-way latency of all packets between two nodes, or over both, e.g., the mean of the queueing delays of a flow on all nodes along a path. The aggregation function can be numerical, e.g., median, sum, minimum, it can be logical, e.g., "true if all are true", "true if at least 50\% of values are true", or an arbitrary function which maps multiple input values to an output value.

Observed property: A property that is observed for a specific path element or path, e.g., using measurements. For example, the one-way delay of a specific packet transmitted from one node to another node can be measured.

Assessed property: An approximate calculation or assessment of the value of a property. An assessed property includes the reliability of the calculation or assessment. The notion of reliability depends on the property. For example, a path property based on an approximate calculation may describe the expected median one-way latency of packets sent on a path within the next second, including the confidence level and interval. A non-numerical assessment may instead include the likelihood that the property holds.

3. Use Cases for Path Properties

When a path-aware network exposes path properties to hosts or other entities, these entities may use this information to achieve different goals. This section lists several use cases for path properties. Note that this is not an exhaustive list, as with every new technology and protocol, novel use cases may emerge, and new path properties may become relevant.

3.1. Performance Monitoring and Enhancement

Network operators can observe path properties (e.g., measured by on-path devices), to monitor Quality of Service (QoS) characteristics of recent end-user traffic on a path or subpath through their network. Such properties may help identify potential performance problems or trigger countermeasures to enhance performance.

3.2. Path Selection

Entities can choose what traffic to send over which path or subset of paths. Entities may select their paths to fulfill a specific goal, e.g., related to security or performance. As an example of security-related path selection, an entity may allow or disallow sending traffic over paths involving specific networks or nodes to enforce traffic policies. In an enterprise network where all traffic has to go through a specific firewall, a path-aware host can implement this policy using path selection, in which case the host needs to be aware of paths involving that firewall. As an example of performance-related path selection, an entity may prefer paths with performance properties that best match its traffic, e.g., retrieving a small webpage as quickly as possible over a path with short One-Way Delays in both directions, or retrieving a large file over a path with high Link Capacities on all links. Note, there may be trade-offs between path properties (e.g., One-Way Delay and Link Capacity), and entities may influence these trade-offs with their choices. As a baseline, a path selection algorithm should aim to not perform worse than the default case most of the time.

Path selection can be done both by hosts and by entities within the network: A network (e.g., an AS) can adjust its path selection for internal or external routing based on the path properties. In BGP, the Multi Exit Discriminator (MED) attribute decides which path to choose if other attributes are equal; in a path aware network, instead of using this single MED value, other properties such as maximum or available/expected data rate could additionally be used to improve load balancing. A host might be able to select between a set of paths, either if there are several paths to the same destination (e.g., if the host is a mobile device with two wireless interfaces, both providing a path), or if there are several destinations, and thus several paths, providing the same service (e.g., Application-Layer Traffic Optimization (ALTO) [RFC5693], an application layer peer-to-peer protocol allowing hosts a better-than-random peer selection). Care needs to be taken when selecting paths based on path properties, as path properties that were previously measured may have become outdated and, thus, useless to predict the path properties of packets sent now.

3.3. Traffic Configuration

When sending traffic over a specific path, entities can adjust this traffic based on the properties of the path. For example, an entity may select an appropriate protocol depending on the capabilities of the on-path devices, or adjust protocol parameters to an existing path. An example of traffic configuration is a video streaming application choosing an (initial) video quality based on the achievable data rate, or the monetary cost to send data across a network, eventually on a given path, using a volume-based or flat-rate cost model.

Conversely, the selection of a protocol may influence the devices that will be involved in a path. For example, a 0-RTT Transport Converter [I-D.ietf-tcpm-converters] will be involved in a path only when invoked by a host; such invocation will lead to the use of MPTCP or TCPinc capabilities while such use is not supported via the default forwarding path. Another example of traffic policies is a connection which may be composed of multiple streams; each stream with specific service requirements. A host may decide to invoke a given service function (e.g., transcoding) only for some streams while others are not processed by that service function.

4. Examples of Path Properties

This Section gives some examples of Path Properties which may be useful, e.g., for the use cases described in Section 3.

Path properties may be relatively dynamic, e.g., the one-way delay of a packet sent over a specific path, or non-dynamic, e.g., the MTU of an ethernet link which only changes infrequently. Usefulness over time differs depending on how dynamic a property is: The merit of a momentary measurement of a dynamic path property diminishes greatly as time goes on, e.g. the merit of an RTT measurement from a few seconds ago is quite small, while a non-dynamic path property might stay relevant for a longer period of time, e.g. a NAT typically stays on a specific path during the lifetime of a connection involving packets sent over this path.

From the point of view of a host, path properties may relate to path elements close to the host, i.e., within the first few hops, or they may include path elements far from the host, e.g. list of ASes traversed. The visibility of path properties to a specific entity may depend on factors such as the physical or network distance or the existence of trust or contractual relationships between the entity and the path element(s).

Furthermore, entities may or may not be able to influence the path elements on their path and their path properties. For example, a user might select between multiple potential adjacent links by selecting between multiple available WiFi Access Points. Or when connected to an Access Point, the user may move closer to enable their device to use a different access technology, potentially increasing the data rate available to the device. Another example is a user changing their data plan to reduce the Monetary Cost to transmit a given amount of data across a network.

Access Technology: The physical or link layer technology used for transmitting or receiving a flow on one or multiple path elements. The Access Technology may be given in an abstract way, e.g., as a WiFi, Wired Ethernet, or Cellular link. It may also be given as a specific technology, e.g., as a 2G, 3G, 4G, or 5G cellular link, or an 802.11a, b, g, n, or ac WiFi link. Other path elements relevant to the access technology may include on-path devices, such as elements of a cellular backbone network. Note that there is no common registry of possible values for this property.

Monetary Cost: The price to be paid to transmit a specific flow across a network to which one or multiple path elements belong.

Service function: A service function that a path element applies to a flow, see [RFC7665]. Examples of abstract service functions include firewalls, Network Address Translation (NAT), and TCP optimizers.

Administrative Domain: The administrative domain, e.g., the ICP area, AS, or Service provider network to which a path element or subpath belongs.

Disjointness: For a set of two paths, the number of shared path elements can be a measure of intersection (e.g., Jaccard coefficient, which is the number of shared elements divided by the total number of elements). Conversely, the number of non-shared path elements can be a measure of disjointness (e.g., 1 - Jaccard coefficient). A multipath protocol might use disjointness of paths as a metric to reduce the number of single points of failure.

Path MTU: The maximum size, in octets, of an IP packet that can be transmitted without fragmentation on a subpath.

Transport Protocols available: Whether a specific transport protocol can be used to establish a connection over a path or subpath. A host may cache its knowledge about recent successfully established connections using specific protocols, e.g., a QUIC connection, or an MPTCP subflow.

Protocol Features available: Whether a specific protocol feature is available over a path or subpath, e.g., Explicit Congestion Notification (ECN), or TCP Fast Open.

Some path properties express the performance of the transmission of a packet or flow over a link or subpath. Such transmission performance properties can be measured or approximated, e.g., by hosts or by path elements on the path. They might be made available in an aggregated form, such as averages or minimums. See [ANRW18-Metrics] for a discussion of how to measure some transmission performance properties at the host. Properties related to a path element which constitutes a single layer 2 domain are abstracted from the used physical and link layer technology, similar to [RFC8175].

Link Capacity: The link capacity is the maximum data rate at which data that was sent over a link can correctly be received at the node adjacent to the link. This property is analogous to the link capacity defined in [RFC5136] but not restricted to IP-layer traffic.

Link Usage: The link usage is the actual data rate at which data that was sent over a link is correctly received at the node adjacent to the link. This property is analogous to the link usage defined in [RFC5136] but not restricted to IP-layer traffic.

One-Way Delay: The one-way delay is the delay between a node sending a packet and another node on the same path receiving the packet. This property is analogous to the one-way delay defined in [RFC7679] but not restricted to IP-layer traffic.

One-Way Delay Variation: The variation of the one-way delays within a flow. This property is similar to the one-way delay variation defined in [RFC3393] but not restricted to IP-layer traffic and defined for packets on the same flow instead of packets sent between a source and destination IP address.

One-Way Packet Loss: Packets sent by a node but not received by another node on the same path after a certain time interval are considered lost. This property is analogous to the one-way loss defined in [RFC7680] but not restricted to IP-layer traffic. Metrics such as loss patterns [RFC3357] and loss episodes [RFC6534] can be expressed as aggregated properties.

5. Security Considerations

If nodes are basing policy or path selection decisions on path properties, they need to rely on the accuracy of path properties that other devices communicate to them. In order to be able to trust such path properties, nodes may need to establish a trust relationship or be able to verify the authenticity, integrity, and correctness of path properties received from another node.

6. IANA Considerations

This document has no IANA actions.

7. Informative References

[ANRW18-Metrics]
Enghardt, T., Tiesel, P., and A. Feldmann, "Metrics for access network selection", Proceedings of the Applied Networking Research Workshop on - ANRW '18, DOI 10.1145/3232755.3232764, 2018.

[I-D.ietf-tcpm-converters]
Bonaventure, O., Boucadair, M., Gundavelli, S., Seo, S., and B. Hesmans, "0-RTT TCP Convert Protocol", draft-ietf-tcpm-converters-13 (work in progress), October 2019.

[I-D.irtf-panrg-questions]
Trammell, B., "Current Open Questions in Path Aware Networking", draft-irtf-panrg-questions-03 (work in progress), October 2019.

[RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, DOI 10.17487/RFC1122, October 1989, <https://www.rfc-editor.org/info/rfc1122>.

[RFC3357] Koodli, R. and R. Ravikanth, "One-way Loss Pattern Sample Metrics", RFC 3357, DOI 10.17487/RFC3357, August 2002, <https://www.rfc-editor.org/info/rfc3357>.

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, DOI 10.17487/RFC3393, November 2002, <https://www.rfc-editor.org/info/rfc3393>.

[RFC5136] Chimento, P. and J. Ishac, "Defining Network Capacity", RFC 5136, DOI 10.17487/RFC5136, February 2008, <https://www.rfc-editor.org/info/rfc5136>.

[RFC5693] Seedorf, J. and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement", RFC 5693, DOI 10.17487/RFC5693, October 2009, <https://www.rfc-editor.org/info/rfc5693>.

[RFC6534] Duffield, N., Morton, A., and J. Sommers, "Loss Episode Metrics for IP Performance Metrics (IPPM)", RFC 6534, DOI 10.17487/RFC6534, May 2012, <https://www.rfc-editor.org/info/rfc6534>.

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[RFC7679] Almes, G., Kalidindi, S., Zekauskas, M., and A. Morton, Ed., "A One-Way Delay Metric for IP Performance Metrics (IPPM)", STD 81, RFC 7679, DOI 10.17487/RFC7679, January 2016, <https://www.rfc-editor.org/info/rfc7679>.

[RFC7680] Almes, G., Kalidindi, S., Zekauskas, M., and A. Morton, Ed., "A One-Way Loss Metric for IP Performance Metrics (IPPM)", STD 82, RFC 7680, DOI 10.17487/RFC7680, January 2016, <https://www.rfc-editor.org/info/rfc7680>.

[RFC8175] Ratliff, S., Jury, S., Satterwhite, D., Taylor, R., and B. Berry, "Dynamic Link Exchange Protocol (DLEP)", RFC 8175, DOI 10.17487/RFC8175, June 2017, <https://www.rfc-editor.org/info/rfc8175>.

Acknowledgments

Thanks to the Path-Aware Networking Research Group for the discussion and feedback. Specifically, thanks to Mohamed Boudacair for the detailed review and various text suggestions, thanks to Brian Trammell for suggesting the flow definition, and thanks to Adrian Perrig and Matthias Rost for the detailed feedback. Thanks to Paul Hoffman for the editorial changes.

Authors' Addresses

Theresa Enghardt
TU Berlin

Email: theresa@inet.tu-berlin.de

Cyrill Kraehenbuehl
ETH Zuerich

Email: cyrill.kraehenbuehl@inf.ethz.ch

Path Aware Networking RG
Internet-Draft
Intended status: Informational
Expires: March 2, 2021

B. Trammell
Google Switzerland GmbH
August 29, 2020

Current Open Questions in Path Aware Networking
draft-irtf-panrg-questions-07

Abstract

In contrast to the present Internet architecture, a path-aware internetworking architecture has two important properties: it exposes the properties of available Internet paths to endpoints, and provides for endpoints and applications to use these properties to select paths through the Internet for their traffic. This document poses questions in path-aware networking open as of 2020, that must be answered in the design, development, and deployment of path-aware internetworks. It was originally written to frame discussions in the Path Aware Networking proposed Research Group (PANRG), and has been published to snapshot current thinking in this space.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction to Path-Aware Networking

In the current Internet architecture, the network layer provides an unverifiable, best-effort service: an application can assume that a packet with a given destination address will eventually be forwarded toward that destination, but little else. A transport layer protocol such as TCP can provide reliability over this best-effort service, and a protocol above the network layer such as IPsec AH [RFC4302] or TLS [RFC8446] can authenticate the remote endpoint. However, little, if any, explicit information about the path is available, and assumptions about that path often do not hold, sometimes with serious impacts on the application, as in the case with BGP hijacking attacks.

By contrast, in a path-aware internetworking architecture, endpoints have the ability to select or influence the path through the network used by any given packet, and the network and transport layers explicitly expose information about the path or paths available from one endpoint to another, and vice versa, to those endpoints and the applications running on them, so that they can make this selection.

Path selection provides transparency and control to applications and users of the network. Selection may be made at either the application layer or the transport layer. Path control at the packet level enables the design of new transport protocols that can leverage multipath connectivity across disjoint paths through the Internet, even over a single physical interface. When exposed to applications, or to end-users through a system configuration interface, path control allows the specification of constraints on the paths that traffic should traverse, for instance to confound passive surveillance in the network core [RFC7624].

We note that this property of "path awareness" already exists in many Internet-connected networks within single domains. Indeed, much of the practice of network engineering using encapsulation at layer 3 can be said to be "path aware", in that it explicitly assigns traffic at tunnel endpoints to a given path within the network. Path-aware internetworking seeks to extend this awareness across domain boundaries without resorting to overlays, except as a transition technology.

This document presents a snapshot of open questions in this space that will need to be answered in order to realize a path-aware internetworking architecture; it is published to further frame discussions within and outside the Path Aware Networking Research Group, and is published with the rough consensus of that group.

1.1. Definition

For purposes of this document, "path aware networking" describes endpoint discovery of the properties of paths they use for communication, and endpoint reaction to these properties that affects routing and/or transmission; note that this can and already does happen to some extent in the current Internet architecture. Expanding on this definition, a "path aware internetwork" is one in which endpoint discovery of path properties and endpoint selection of paths used by traffic exchanged by the endpoint are explicitly supported, regardless of the specific design of the protocol features which enable this discovery and selection.

Research into path aware networking covers any and all aspects of designing, building, and operating path aware internetworks or the networks and endpoints attached to them. This document presents a collection of research questions to address in order to make a path aware Internet a reality.

2. Questions

Realizing path-aware networking requires answers to a set of open research questions. This document poses these questions, as a starting point for discussions about how to realize path awareness in the Internet, and to direct future research efforts within the Path Aware Networking Research Group.

2.1. A Vocabulary of Path Properties

In order for information about paths to be exposed to an endpoint, and for the endpoint to make use of that information, it is necessary to define a common vocabulary for paths through an internetwork, and properties of those paths. The elements of this vocabulary could include terminology for components of a path and properties defined for these components, for the entire path, or for subpaths of a path. These properties may be relatively static, such as the presence of a given node or service function on the path; as well as relatively dynamic, such as the current values of metrics such as loss and latency.

This vocabulary must be defined carefully, as its design will have impacts on the expressiveness of a given path-aware internetworking architecture. This expressiveness also exhibits tradeoffs. For example, a system that exposes node-level information for the topology through each network would maximize information about the individual components of the path at the endpoints, at the expense of making internal network topology universally public, which may be in conflict with the business goals of each network's operator. Furthermore, properties related to individual components of the path may change frequently and may quickly become outdated. However, aggregating the properties of individual components to distill end-to-end properties for the entire path is not trivial.

The first question: how are paths and path properties defined and represented?

2.2. Discovery, Distribution, and Trustworthiness of Path Properties

Once endpoints and networks have a shared vocabulary for expressing path properties, the network must have some method for distributing those path properties to the endpoint. Regardless of how path property information is distributed to the endpoints, the endpoints require a method to authenticate the properties - to determine that they originated from and pertain to the path that they purport to.

Choices in distribution and authentication methods will have impacts on the scalability of a path-aware architecture. Possible dimensions in the space of distribution methods include in-band versus out-of-band, push versus pull versus publish-subscribe, and so on. There are temporal issues with path property dissemination as well, especially with dynamic properties, since the measurement or elicitation of dynamic properties may be outdated by the time that information is available at the endpoints, and interactions between the measurement and dissemination delay may exhibit pathological behavior for unlucky points in the parameter space.

The second question: how do endpoints and applications get access to trustworthy path properties?

2.3. Supporting Path Selection

Access to trustworthy path properties is only half of the challenge in establishing a path-aware architecture. Endpoints must be able to use this information in order to select paths for specific traffic they send. As with the dissemination of path properties, choices made in path selection methods will also have an impact on the tradeoff between scalability and expressiveness of a path-aware architecture. One key choice here is between in-band and out-of-band control of path selection. Another is granularity of path selection (whether per packet, per flow, or per larger aggregate), which also has a large impact on the scalabilty/expressiveness tradeoff. Path selection must, like path property information, be trustworthy, such that the result of a path selection at an endpoint is predictable. Moreover, any path selection mechanism should aim to provide an outcome that is not worse than using a single path, or selecting paths at random.

The third question: how can endpoints select paths to use for traffic in a way that can be trusted by the network, the endpoints, and the applications using them?

2.4. Interfaces for Path Awareness

In order for applications to make effective use of a path-aware networking architecture, the control interfaces presented by the network and transport layers must also expose path properties to the application in a useful way, and provide a useful set of paths among which the application can select. Path selection must be possible based not only on the preferences and policies of the application developer, but of end-users as well. Also, the path selection interfaces presented to applications and end users will need to support multiple levels of granularity. Most applications' requirements can be satisfied with the expression path selection policies in terms of properties of the paths, while some applications may need finer-grained, per-path control. These interfaces will need to support incremental development and deployment of applications, and provide sensible defaults, to avoid hindering their adoption.

The fourth question: how can interfaces to the transport and application layers support the use of path awareness?

2.5. Implications of Path Awareness for the Data Plane

In the current Internet, the basic assumption that at a given time all traffic for a given flow will receive the same network treatment and traverse the same path or equivalend paths often holds. In a path aware network, this assumption is more easily violated. The weakening of this assumption has implications for the design of protocols above any path-aware network layer.

For example, one advantage of multipath communication is that a given end-to-end flow can be "sprayed" along multiple paths in order to confound attempts to collect data or metadata from those flows for pervasive surveillance purposes [RFC7624]. However, the benefits of this approach are reduced if the upper-layer protocols use linkable identifiers on packets belonging to the same flow across different paths. Clients may mitigate linkability by opting to not re-use cleartext connection identifiers, such as TLS session IDs or tickets, on separate paths. The privacy-conscious strategies required for effective privacy in a path-aware Internet are only possible if higher-layer protocols such as TLS permit clients to obtain unlinkable identifiers.

The fifth question: how should transport-layer and higher layer protocols be redesigned to work most effectively over a path-aware networking layer?

2.6. What is an Endpoint?

The vision of path-aware networking articulated so far makes an assumption that path properties will be disseminated to endpoints on which applications are running (terminals with user agents, servers, and so on). However, incremental deployment may require that a path-aware network "core" be used to interconnect islands of legacy protocol networks. In these cases, it is the gateways, not the application endpoints, that receive path properties and make path selections for that traffic. The interfaces provided by this gateway are necessarily different than those a path-aware networking layer provides to its transport and application layers, and the path property information the gateway needs and makes available over those interfaces may also be different.

The sixth question: how is path awareness (in terms of vocabulary and interfaces) different when applied to tunnel and overlay endpoints?

2.7. Operating a Path Aware Network

The network operations model in the current Internet architecture assumes that traffic flows are controlled by the decisions and policies made by network operators, as expressed in interdomain and intradomain routing protocols. In a network providing path selection to the endpoints, however, this assumption no longer holds, as endpoints may react to path properties by selecting alternate paths. Competing control inputs from path-aware endpoints and the routing control plane may lead to more difficult traffic engineering or nonconvergent forwarding, especially if the endpoints' and operators' notion of the "best" path for given traffic diverges significantly. The degree of difficulty may depend on the fidelity of information made available to path selection algorithms at the endpoints. Explicit path selection can also specify outbound paths, while BGP policies are expressed in terms of inbound traffic.

A concept for path aware network operations will need to have clear methods for the resolution of apparent (if not actual) conflicts of intent between the network's operator and the path selection at an endpoint. It will also need set of safety principles to ensure that increasing path control does not lead to decreasing connectivity; one such safety principle could be "the existence of at least one path between two endpoints guarantees the selection of at least one path between those endpoints."

The seventh question: how can a path aware network in a path aware internetwork be effectively operated, given control inputs from the network administrator as well as from the endpoints?

2.8. Deploying a Path Aware Network

The vision presented in the introduction discusses path aware networking from the point of view of the benefits accruing at the endpoints, to designers of transport protocols and applications as well as to the end users of those applications. However, this vision requires action not only at the endpoints but within the interconnected networks offering path aware connectivity. While the specific actions required are a matter of the design and implementation of a specific realization of a path aware protocol stack, it is clear than any path aware architecture will require network operators to give up some control of their networks over to endpoint-driven control inputs.

Here the question of apparent versus actual conflicts of intent arises again: certain network operations requirements may appear essential, but are merely accidents of the interfaces provided by current routing and management protocols. Incentives for deployment must show how existing network operations requirements are met through new path selection and property dissemination mechanisms.

The incentives for network operators and equipment vendors need to be made clear, in terms of a plan to transition [RFC8170] an internetwork to path-aware operation, one network and facility at a time. This plan to transition must also take into account that the dynamics of path aware networking early in this transition (when few endpoints and flows in the Internet use path selection) may be different than those later in the transition.

The eighth question: how can the incentives of network operators and end-users be aligned to realize the vision of path aware networking, and how can the transition from current ("path-oblivious") to path-aware networking be managed?

3. Acknowledgments

Many thanks to Adrian Perrig, Jean-Pierre Smith, Mirja Kuehlewind, Olivier Bonaventure, Martin Thomson, Shwetha Bhandari, Chris Wood, Lee Howard, Mohamed Boucadair, Thorben Krueger, Gorry Fairhurst, Spencer Dawkins, and Theresa Enghardt for discussions leading to questions in this document, and for feedback on the document itself.

This work is partially supported by the European Commission under Horizon 2020 grant agreement no. 688421 Measurement and Architecture for a Middleboxed Internet (MAMI), and by the Swiss State Secretariat for Education, Research, and Innovation under contract no. 15.0268. This support does not imply endorsement.

4. Informative References

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <https://www.rfc-editor.org/info/rfc4302>.

[RFC7624] Barnes, R., Schneier, B., Jennings, C., Hardie, T., Trammell, B., Huitema, C., and D. Borkmann, "Confidentiality in the Face of Pervasive Surveillance: A Threat Model and Problem Statement", RFC 7624, DOI 10.17487/RFC7624, August 2015, <https://www.rfc-editor.org/info/rfc7624>.

[RFC8170] Thaler, D., Ed., "Planning for Protocol Adoption and Subsequent Transitions", RFC 8170, DOI 10.17487/RFC8170, May 2017, <https://www.rfc-editor.org/info/rfc8170>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

Author's Address

Brian Trammell
Google Switzerland GmbH
Gustav-Gull-Platz 1
8004 Zurich
Switzerland

Email: ietf@trammell.ch

PANRG
Internet-Draft
Intended status: Informational
Expires: April 18, 2019

S. Dawkins, Ed.
Huawei Technologies
October 15, 2018

Path Aware Networking: Obstacles to Deployment (A Bestiary of Roads Not Taken)

draft-irtf-panrg-what-not-to-do-00

Abstract

At the first meeting of the Path Aware Networking Research Group, Oliver Bonaventure led a discussion of mostly-unsuccessful attempts to exploit Path Awareness to achieve a variety of goals, for a variety of reasons, over the past decade. At the end of that discussion, the research group agreed to catalog and analyze these ideas, in order to extract insights and lessons for path-aware networking researchers.

This document contains that catalog and analysis.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

At IETF 99, the Path Aware Networking Research Group [PANRG] held its first meeting [PANRG-99], and the first presentation in that session was "A Decade of Path Awareness" [PATH-Decade]. At the end of this discussion, two things were abundantly clear.

- o The Internet community has accumulated considerable experience with many Path Awareness ideas over a long period of time, and

- o Although some Path Awareness ideas have been successfully deployed (for example, Differentiated Services, or DiffServ [RFC2475]), most of these ideas haven't seen widespread adoption. The reasons for non-adoption are many, and are worthy of study.

The meta-lessons from that experience were

- o Path Aware Networking is more Research than Engineering, so establishing an IRTF Research Group for Path Aware Networking is the right thing to do [RFC7418].

- o Analyzing a catalog of past experience to learn the reasons for non-adoption would be a great first step for the Research Group.

Allison Mankin, the IRTF Chair, officially chartered the Path Aware Networking Research Group in July, 2018.

This document contains the analysis performed by that research group (see Section 2), based on that catalog (see Section 4).

1.1. About this Document

This document is not intended to catalog every idea about Path Aware Networking that we can find. Instead, we include enough ideas to provide background for new lessons to guide researchers in their work, in order to add those lessons to Section 2.

There is no shame to having an idea included in this document. As shown in Section 2, the quality of specific proposals had little to do with whether they were deployed or not. The first contribution added to this document was for a proposal from the editor of this document Section 4.3, and it wasn't deployed. When these proposals were made, the proponents were trying to engineer something when they should have been trying to research it. Actual shame would be failing to learn from experience, and failing to share that experience with other networking researchers and engineers.

We may stand on the shoulders of giants, but most of those giants' Path Aware Networking ideas weren't deployed, either!

Discussion of specific contributed experiences and this document in general should take place on the PANRG mailing list.

1.2. A Note for the Editor

(Remove after taking these actions)

The to-do list for upcoming revisions includes

- o Confirm that the Summary of Lessons Learned makes sense and is complete, in consultation with the Research Group.

- o If the Research Group identifies technologies that provided lessons that aren't included in Section 2, solicit contributions for those technologies.

- o Edit the contributed subsections for basic consistency (since they have different contributors providing initial material).

1.3. Architectural Guidance

As background for understanding the Lessons Learned contained in this document, the reader is encouraged to become familiar with the Internet Architecture Board's documents on "What Makes for a Successful Protocol?" [RFC5218] and "Planning for Protocol Adoption and Subsequent Transitions" [RFC8170].

Although these two documents do not specifically target path-aware networking protocols, they are helpful resources on successful protocol adoption and deployment.

Because there is an economic aspect to decisions about deployment, the IAB Workshop on Internet Technology Adoption and Transition [ITAT] report [RFC7305] also provides food for thought.

2. Summary of Lessons Learned

This section summarizes the Lessons Learned from the contributed sections in Section 4.

Each Lesson Learned is tagged with one or more contributions that encountered this obstacle as a significant impediment to deployment. Other contributed technologies may have also encountered this obstacle, but this obstacle may not have been the biggest impediment to deployment.

- o The benefit of Path Awareness has to be great enough to overcome entropy for already-deployed devices. The colloquial American English expression, "If it ain't broke, don't fix it" is a "best current practice" on today's Internet. (See Section 4.2 and Section 4.3).

- o Providing benefits for early adopters is key - if everyone must deploy a technology in order for the topology to provide benefits, or even to work at all, the technology is unlikely to be adopted. (See Section 4.1 and Section 4.2).

- o "Follow the money." If operators can't charge for a Path Aware technology in order to recover the costs of deploying it, the benefits to the operator must be really significant. (See Section 4.3).

- Impact of a Path Aware technology on operational practices can prevent deployment of promising technology. (See Section 4.4).

- Per-connection state in intermediate devices is an impediment to adoption and deployment. (See Section 4.1).

- Modern routers aren't designed to make heavy use of in-band signaling using mechanisms such as IPv4 and IPv6 Router Alert Options (RAO), so operators are reluctant to deploy technologies that rely on these signals. (See Section 4.5).

- If endpoints can't be trusted, operators are reluctant to deploy technologies that rely on endpoints sending unauthenticated control signals to routers. (See Section 4.5).

- If intermediate devices along the path can't be trusted, it's unlikely that endpoints will rely on signals from intermediate devices to drive changes to endpoint behaviors. (See Section 4.3).

- The Internet is a distributed system, so the more a technology relies on information propagated from distant hosts and routers, the less likely that information is to be accurate. (See Section 4.2).

- Transport protocol technologies may require information from applications, in order to work effectively, but applications may not know the information they need to provide. (See Section 4.2).

3. Template for Contributions

There are many things that could be said about the Path Aware networking technologies that have been developed. For the purposes of this document, contributors are requested to provide

- the name of a technology, including an abbreviation if one was used

- if available, a long-term pointer to the best reference describing the technology

- a short description of the problem the technology was intended to solve

- a short description of the reasons why the technology wasn't adopted

- a short statement of the lessons that researchers can learn from our experience with this technology.

This document is being built collaboratively. To contribute your experience, please send a Github pull request to https://github.com/panrg/draft-dawkins-panrg-what-not-to-do.

4. Contributions

Additional contributions that provide Lessons Learned beyond those already captured in Section 2 are welcomed.

4.1. Integrated Services (IntServ)

The suggested references for IntServ are:

- RFC 1633 Integrated Services in the Internet Architecture: an Overview [RFC1633]

- RFC 2211 Specification of the Controlled-Load Network Element Service [RFC2211]

- RFC 2212 Specification of Guaranteed Quality of Service [RFC2212]

- RFC 2215 General Characterization Parameters for Integrated Service Network Elements [RFC2215]

- RFC 2205 Resource ReSerVation Protocol (RSVP) [RFC2205]

In 1994, when the IntServ architecture document [RFC1633] was published, real-time traffic was first appearing on the Internet. At that time, bandwidth was a scarce commodity. Internet Service Providers built networks over DS3 (45 Mbps) infrastructure, and sub-rate (< 1 Mpbs) access was common. Therefore, the IETF anticipated a need for a fine-grained QoS mechanism.

In the IntServ architecture, some applications require service guarantees. Therefore, those applications use the Resource Reservation Protocol (RSVP) [RFC2205] to signal bandwidth reservations across the network. Every router in the network maintains per-flow state in order to a) perform call admission control and b) deliver guaranteed service.

Applications use Flow Specification (Flow Specs) [RFC2210] to describe the traffic that they emit. RSVP reserves bandwidth for traffic on a per Flow Spec basis.

4.1.1. Reasons for Non-deployment

IntServ was never widely deployed because of its cost. The following factors contributed to cost:

- o IntServ must be deployed on every router within the QoS domain
- o IntServ maintained per flow state

As IntServ was being discussed, the following occurred:

- o It became more cost effective to solve the QoS problem by adding bandwidth. Between 1994 and 2000, Internet Service Providers upgraded their infrastructures from DS3 (45 Mbps) to OC-48 (2.4 Gbps). This meant that even if an endpoint was using IntServ in an IntServ-enabled network, its requests would never be denied, so endpoints and Internet Service Providers had little reason to enable IntServ.
- o DiffServ [RFC2475] offered a more cost-effective, albeit less fine-grained, solution to the QoS problem.

4.1.2. Lessons Learned.

The following lessons were learned:

- o Any mechanism that requires a router to maintain per-flow state is not likely to succeed.
- o Any mechanism that requires an operator to upgrade all of its routers is not likely to succeed.

IntServ was never widely deployed. However, the technology that it produced was deployed for reasons other than bandwidth management. RSVP is widely deployed as an MPLS signaling mechanism. BGP uses Flow Specs to distribute firewall filters.

4.2. Quick-Start TCP

The suggested references for Quick-Start TCP are:

- o RFC 4782 Quick-Start for TCP and IP [RFC4782]

- o Determining an appropriate sending rate over an underutilized network path [SAF07]

- o Fast Startup Internet Congestion Control for Broadband Interactive Applications [Sch11]

Quick-Start [RFC4782] is an experimental TCP extension that leverages support from the routers on the path to determine an allowed sending rate, either at the start of data transfers or after idle periods. In these cases, a TCP sender cannot easily determine an appropriate sending rate, given the lack of information about the path. The default TCP congestion control therefore uses the time-consuming slow-start algorithm. With Quick-Start, connections are allowed to use higher sending rates if there is significant unused bandwidth along the path, and if the sender and all of the routers along the path approve the request. By examining Time To Live (TTL) fields, a sender can determine if all routers have approved the Quick-Start request. The protocol also includes a nonce that provides protection against cheating routers and receivers. If the Quick-Start request is explicitly approved by all routers along the path, the TCP host can send at up to the approved rate; otherwise TCP would use the default congestion control. Quick-Start requires modifications in the involved end-systems as well in routers. Due to the resulting deployment challenges, Quick-Start was only proposed in [RFC4782] for controlled environments.

The Quick-Start protocol is a lightweight, coarse-grained, in-band, network-assisted fast startup mechanism. The benefits are studied by simulation in a research paper [SAF07] that complements the protocol specification. The study confirms that Quick-Start can significantly speed up mid-sized data transfers. That paper also presents router algorithms that do not require keeping per-flow state. Later studies [Sch11] comprehensively analyzes Quick-Start with a full Linux implementation and with a router fast path prototype using a network processor. In both cases, Quick-Start could be implemented with limited additional complexity.

4.2.1. Reasons for Non-deployment

However, the experiments with Quick-Start in [Sch11] reveal several challenges:

- o Having information from the routers along the path can reduce the risk of congestion, but cannot avoid it entirely. Determining whether there is unused capacity is not trivial in actual router and host implementations. Data about available bandwidth visible at the IP layer may be imprecise, and due to the propagation delay, information can already be outdated when it reaches the sender. There is a trade-off between the speedup of data transfers and the risk of congestion even with Quick-Start.

- For scalable router fast path implementation, it is important to enable parallel processing of packets, as this is a widely used method e.g. in network processors. One challenge is synchronization of information between different packets, which should be avoided as much as possible.

- Only selected applications can benefit from Quick-Start. For achieving an overall benefit, it is important that senders avoid sending unnecessary Quick-Start requests, e.g. for connections that will only send a small amount of data. This typically requires application-internal knowledge. It is a mostly unsolved question how a sender can indeed determine the data rate that Quick-Start shall request for.

After completion of the Quick-Start specification, there have been large-scale experiments with an initial window of up to 10 MSS [RFC6928]. This alternative "IW10" approach can also ramp up data transfers faster than the standard TCP congestion control, but it only requires sender-side TCP modifications. As a result, this approach can be easier and incrementally deployed in the Internet. While theoretically Quick-Start can outperform "IW10", the absolute improvement of data transfer times is rather small in many cases. After publication of [RFC6928], most modern TCP stacks have increased their default initial window. There is no known deployment of Quick-Start TCP.

4.2.2. Lessons Learned

There are some lessons learned from Quick-Start. Despite being a very light-weight protocol, Quick-Start suffers from poor incremental deployment properties, both regarding the required modifications in network infrastructure as well as its interactions with applications. Except for corner cases, congestion control can be quite efficiently performed end-to-end in the Internet, and in modern TCP stacks there is not much room for significant improvement by additional network support.

4.3. Triggers for Transport (TRIGTRAN)

The suggested references for TRIGTRAN are:

- TRIGTRAN BOF at IETF 55 [TRIGTRAN-55]

- TRIGTRAN BOF at IETF 56 [TRIGTRAN-56]

TCP [RFC0793] has a well-known weakness - the end-to-end flow control mechanism has only a single signal, the loss of a segment, and semi-modern TCPs (since the late 1980s) have interpreted the loss of a segment as evidence that the path between two endpoints has become congested enough to exhaust buffers on intermediate hops, so that the TCP sender should "back off" - reduce its sending rate until it knows that its segments are now being delivered without loss [RFC2581]. More modern TCPs have added a growing array of strategies about how to establish the sending rate [RFC5681], but when a path is no longer operational, TCPs can wait many seconds before retrying a segment, even if the path becomes operational while the sender is waiting for its next retry.

The thinking in Triggers for Transport was that if a path completely stopped working because its first-hop link was "down", that somehow TCP could be signaled when the first-hop link returned to service, and the sending TCP could retry immediately, without waiting for a full Retransmission Time Out (RTO) period.

4.3.1. Reasons for Non-deployment

Two TRIGTRAN BOFs were held, at IETF 55 [TRIGTRAN-55] and IETF 56 [TRIGTRAN-56], but this work was not chartered, and there was no interest in deploying TRIGTRAN unless it was chartered and standardized in the IETF.

4.3.2. Lessons Learned.

The reasons why this work was not chartered provide several useful lessons for researchers.

- TRIGTRAN triggers are only provided when the first-hop link is "down", so TRIGTRAN triggers couldn't replace normal TCP retransmission behavior if the path failed because some link further along the network path was "down". So TRIGTRAN triggers added complexity to an already complex TCP state machine, and didn't allow any existing complexity to be removed.

- The state of the art in the early 2000s was that TRIGTRAN triggers were assumed to be unauthenticated, so they couldn't be trusted to tell a sender to "speed up", only to "slow down". This reduced the potential benefit to implementers.

- intermediate forwarding devices required modification to provide TRIGTRAN triggers, but operators couldn't charge for TRIGTRAN triggers, so there was no way to recover the cost of modifying, testing, and deploying updated intermediate devices.

4.4. Shim6

The suggested references for Shim6 are:

- RFC5533 Shim6: Level 3 Multihoming Shim Protocol for IPv6 [RFC5533]

The IPv6 routing architecture [RFC1887] assumed that most sites on the Internet would be identified by Provider Assigned IPv6 prefixes, so that Default-Free Zone routers only contained routes to other providers, resulting in a very small routing table.

For a single-homed site, this could work well. A multi-homed site with only one upstream provider could also work well, although BGP multihoming from a single upstream provider was often a premium service (costing more than twice as much as two single-homed sites), and if the single upstream provider went out of service, all of the multi-homed paths could fail simultaneously.

IPv4 sites often multihomed by obtaining Provider Independent prefixes, and advertising these prefixes through multiple upstream providers. With the assumption that any multihomed IPv4 site would also multihome in IPv6, it seemed likely that IPv6 routing would be subject to the same pressures to announce Provider Independent prefixes, resulting in a global IPv6 routing table that exhibited the same problems as the global IPv4 routing table. During the early 2000s, work began on a protocol that would provide the same benefits for multihomed IPv6 sites without requiring sites to advertise Provider Independent prefixes into the global routing table.

This protocol, called Shim6, allowed two endpoints to exchange multiple addresses ("Locators") that all mapped to the same endpoint ("Identity"). After an endpoint learned multiple Locators for the other endpoint, it could send to any of those Locators with the expectation that those packets would all be delivered to the endpoint with the same Identity. Shim6 was an example of an "Identity/Locator Split" protocol.

Shim6, as defined in [RFC5533] and related RFCs, provided a workable solution for IPv6 multihoming using Provider Assigned prefixes, including capability discovery and negotiation, and allowing end-to-end application communication to continue even in the face of path failure, because applications don't see Locator failures, and continue to communicate with the same Identity using a different Locator.

4.4.1. Reasons for Non-deployment

Note that the problem being addressed was "site multihoming", but Shim6 was providing "host multihoming". That meant that the decision about what path would be used was under host control, not under router control.

Although more work could have been done to provide a better technical solution, the biggest impediments to Shim6 deployment were operational and business considerations. These impediments were discussed at multiple network operator group meetings, including [Shim6-35] at [NANOG-35].

The technology issues centered around scaling concerns that Shim6 relied on the host to track all the TCP connections and the file descriptions with associated HTTP state, while also tracking Identity/Locator mappings in the kernel, and tracking failures to recognize that a backup path has failed.

The operator issues centered around concerns that operators were performing traffic engineering, but would have no visibility or control over hosts when they chose to begin using another path, and relying on hosts to engineer traffic exposed their networks to oscillation based on feedback loops, as hosts move from path to path. At a minimum, traffic engineering policies must be pushed down to individual hosts. In addition, the usual concerns about firewalls that expected to find a transport-level protocol header in the IP payload, and won't be able to perform firewalling functions because its processing logic would have to look past the Identity header.

The business issues centered removing or reducing the ability to sell BGP multihoming service, which is often more expensive than single-homed connectivity.

4.4.2. Lessons Learned

It is extremely important to take operational concerns into account when a path-aware protocol is making decisions about path selection that may conflict with existing operational practices and business considerations.

We also note that some path-aware networking ideas recycle. Stream Control Transmission Protocol (SCTP) has provided support for multihoming since 2000 [RFC2960], but was designed to transport PSTN signaling messages over IP networks. SCTP was capable of broader applications, but because multi-homed hosts in the 1990s were uncommon, and deployment of new transport protocols such as SCTP required either operating system kernel support or access to raw IP packets until a UDP encapsulation for SCTP [RFC6951] was produced in 2013, SCTP multihoming did not stir up the same operator concerns that Shim6 encountered. Although Shim6 did not achieve significant deployment, the IETF chartered a working group to specify "Multipath TCP" [MP-TCP] in 2009, and Multipath TCP allows general-purpose TCP applications to control path selection, with many of the same advantages and disadvantages of Shim6.

4.5. Next Steps in Signaling (NSIS)

The suggested references for NSIS are:

- o the concluded working group charter [NSIS-CHARTER-2001]
- o RFC 5971 GIST: General Internet Signalling Transport [RFC5971]
- o RFC 5973 NAT/Firewall NSIS Signaling Layer Protocol (NSLP) [RFC5973]
- o RFC 5974 NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling [RFC5974]
- o RFC 5981 "Authorization for NSIS Signaling Layer Protocols [RFC5981]

The Next Steps in Signaling (NSIS) Working Group worked on signaling technologies for network layer resources (e.g., QoS resource reservations, Firewall and NAT traversal).

When RSVP [RFC2205] was used in deployments, a number of questions came up about its perceived limitations and potential missing features. The issues noted in the NSIS Working Group charter [NSIS-CHARTER-2001] include interworking between domains with different QoS architectures, mobility and roaming for IP interfaces, and complexity. Later, the lack of security in RSVP was also recognized ([RFC4094]).

The NSIS Working Group was chartered to tackle those issues and initially focused on QoS signaling as its primary use case. However, over time a new approach evolved that introduced a modular architecture using application-specific signaling protocols (the NSIS Signaling Layer Protocol (NSLP)) on top of a generic signaling transport protocol (the NSIS Transport Layer Protocol (NTLP)).

The NTLP is defined in [RFC5971]. Two NSLPs are defined: the NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling [RFC5974] as well as the NAT/Firewall NSIS Signaling Layer Protocol (NSLP) [RFC5973].

4.5.1. Reasons for Non-deployment

The obstacles for deployment can be grouped into implementation-related aspects and operational aspects.

- Implementation-related aspects:

Although NSIS provides benefits with respect to flexibility, mobility, and security compared to other network signaling technologies, hardware vendors were reluctant to deploy this solution, because it would require additional implementation effort and would result in additional complexity for router implementations.

The NTLP mainly operates as path-coupled signaling protocol, i.e, its messages are processed at the intermediate node's control plane that are also forwarding the data flows. This requires a mechanism to intercept signaling packets while they are forwarded in the same manner (especially along the same path) as data packets. One reason for the non-deployment of NSIS is the usage of the IPv4 and IPv6 Router Alert Option (RAO) to allow for an efficient interception of those path-coupled signaling messages: This option requires router implementations to correctly understand and implement the handling of RAOs, e.g., to only process packet with RAOs of interest and to leave packets with irrelevant RAOs in the fast forwarding processing path (a comprehensive discussion of these issues can be found in [RFC6398]). The latter was an issue with some router implementations at the time of standardization.

Another reason is that path-coupled signaling protocols that interact with routers and request manipulation of state at these routers (or any other network element in general) are under scrutiny: a packet (or sequence of packets) out of the mainly untrusted data path is requesting creation and manipulation of network state. This is seen as potentially dangerous (e.g., opens up a Denial of Service (DoS) threat to a router's control plane) and difficult for an operator to control. End-to-end signaling approaches were considered problematic (see also section 3 of [RFC6398]). There are recommendations on how to secure NSIS nodes and deployments (e.g., [RFC5981]).

- Operational Aspects:

End-to-end signaling technologies not only require trust between customers and their provider, but also among different providers. Especially, QoS signaling technologies would require some kind of dynamic service level agreement support that would imply (potentially quite complex) bilateral negotiations between different Internet service providers. This complexity was currently not considered to be justified and increasing the bandwidth capacity (and thus avoiding bottlenecks) was cheaper than actively managing network resource bottlenecks by using path-coupled QoS signaling technologies. Furthermore, an end-to-end path typically involves several provider domains and these providers need to closely cooperate in cases of failures.

4.5.2. Lessons Learned

One goal of NSIS was to decrease the complexity of the signaling protocol, but a path-coupled signaling protocol comes with the intrinsic complexity of IP-based networks, beyond the complexity of the signaling protocol itself. Sources of intrinsic complexity include

- o the presence of asymmetric routes between endpoints and routers
- o the lack of security and trust at large in the Internet infrastructure
- o the presence of different trust boundaries,
- o the effects of best-effort networks (e.g., packet loss)
- o divergence from the fate sharing principle (e.g., state within the network).

Any path-coupled signaling protocol has to deal with these realities.

Operators view the use of IPv4 and IPv6 Router Alert Option (RAO) to signal routers along the path from end systems with suspicion, because these end systems are usually not authenticated and heavy use of RAOs can easily increase the CPU load on routers that are designed to process most packets using a hardware "fast path".

5. Security Considerations

This document describes ideas that were not adopted and widely deployed on the Internet, so it doesn't affect the security of the Internet.

If this document meets its goals, we may develop new ideas for Path Aware Networking that would affect the security of the Internet, but security considerations for those ideas will be described in the corresponding RFCs that propose them.

6. IANA Considerations

This document makes no requests of IANA.

7. Acknowledgments

Initial material for Section 4.1 on IntServ was provided by Ron Bonica.

Initial material for Section 4.2 on Quick-Start TCP was provided by Michael Scharf.

Initial material for Section 4.3 on Triggers for Transport (TRIGTRAN) was provided by Spencer Dawkins.

Section 4.4 on Shim6 builds on initial material describing obstacles provided by Erik Nordmark, with background added by Spencer Dawkins.

Initial material for Section 4.5 on Next Steps In Signaling (NSIS) was provided by Roland Bless and Martin Stiemerling.

Our thanks to Roland Bless, Ruediger Geib, and Joe Touch, who provided review comments on previous versions.

8. Informative References

[ITAT] "IAB Workshop on Internet Technology Adoption and Transition (ITAT)", December 2013, <https://www.iab.org/activities/workshops/itat/>.

[MP-TCP] "Multipath TCP Working Group Home Page", n.d., <https://datatracker.ietf.org/wg/mptcp/about/>.

[NANOG-35] "North American Network Operators Group NANOG-35 Agenda", October 2005, <https://www.nanog.org/meetings/nanog35/agenda>.

[NSIS-CHARTER-2001] "Next Steps In Signaling Working Group Charter", March 2011, <https://datatracker.ietf.org/doc/charter-ietf-nsis/>.

[PANRG] "Path Aware Networking Research Group (Home Page)", n.d., <https://irtf.org/panrg>.

[PANRG-99]
"Path Aware Networking Research Group - IETF-99", July 2017, <https://datatracker.ietf.org/meeting/99/sessions/panrg>.

[PATH-Decade]
Bonaventure, O., "A Decade of Path Awareness", July 2017, <https://datatracker.ietf.org/doc/slides-99-panrg-a-decade-of-path-awareness/>.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <https://www.rfc-editor.org/info/rfc793>.

[RFC1633] Braden, R., Clark, D., and S. Shenker, "Integrated Services in the Internet Architecture: an Overview", RFC 1633, DOI 10.17487/RFC1633, June 1994, <https://www.rfc-editor.org/info/rfc1633>.

[RFC1887] Rekhter, Y., Ed. and T. Li, Ed., "An Architecture for IPv6 Unicast Address Allocation", RFC 1887, DOI 10.17487/RFC1887, December 1995, <https://www.rfc-editor.org/info/rfc1887>.

[RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <https://www.rfc-editor.org/info/rfc2205>.

[RFC2210] Wroclawski, J., "The Use of RSVP with IETF Integrated Services", RFC 2210, DOI 10.17487/RFC2210, September 1997, <https://www.rfc-editor.org/info/rfc2210>.

[RFC2211] Wroclawski, J., "Specification of the Controlled-Load Network Element Service", RFC 2211, DOI 10.17487/RFC2211, September 1997, <https://www.rfc-editor.org/info/rfc2211>.

[RFC2212] Shenker, S., Partridge, C., and R. Guerin, "Specification of Guaranteed Quality of Service", RFC 2212, DOI 10.17487/RFC2212, September 1997, <https://www.rfc-editor.org/info/rfc2212>.

[RFC2215] Shenker, S. and J. Wroclawski, "General Characterization Parameters for Integrated Service Network Elements", RFC 2215, DOI 10.17487/RFC2215, September 1997, <https://www.rfc-editor.org/info/rfc2215>.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <https://www.rfc-editor.org/info/rfc2475>.

[RFC2581] Allman, M., Paxson, V., and W. Stevens, "TCP Congestion Control", RFC 2581, DOI 10.17487/RFC2581, April 1999, <https://www.rfc-editor.org/info/rfc2581>.

[RFC2960] Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L., and V. Paxson, "Stream Control Transmission Protocol", RFC 2960, DOI 10.17487/RFC2960, October 2000, <https://www.rfc-editor.org/info/rfc2960>.

[RFC4094] Manner, J. and X. Fu, "Analysis of Existing Quality-of-Service Signaling Protocols", RFC 4094, DOI 10.17487/RFC4094, May 2005, <https://www.rfc-editor.org/info/rfc4094>.

[RFC4782] Floyd, S., Allman, M., Jain, A., and P. Sarolahti, "Quick-Start for TCP and IP", RFC 4782, DOI 10.17487/RFC4782, January 2007, <https://www.rfc-editor.org/info/rfc4782>.

[RFC5218] Thaler, D. and B. Aboba, "What Makes for a Successful Protocol?", RFC 5218, DOI 10.17487/RFC5218, July 2008, <https://www.rfc-editor.org/info/rfc5218>.

[RFC5533] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming Shim Protocol for IPv6", RFC 5533, DOI 10.17487/RFC5533, June 2009, <https://www.rfc-editor.org/info/rfc5533>.

[RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, DOI 10.17487/RFC5681, September 2009, <https://www.rfc-editor.org/info/rfc5681>.

[RFC5971] Schulzrinne, H. and R. Hancock, "GIST: General Internet Signalling Transport", RFC 5971, DOI 10.17487/RFC5971, October 2010, <https://www.rfc-editor.org/info/rfc5971>.

[RFC5973] Stiemerling, M., Tschofenig, H., Aoun, C., and E. Davies, "NAT/Firewall NSIS Signaling Layer Protocol (NSLP)", RFC 5973, DOI 10.17487/RFC5973, October 2010, <https://www.rfc-editor.org/info/rfc5973>.

[RFC5974] Manner, J., Karagiannis, G., and A. McDonald, "NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling", RFC 5974, DOI 10.17487/RFC5974, October 2010, <https://www.rfc-editor.org/info/rfc5974>.

[RFC5981] Manner, J., Stiemerling, M., Tschofenig, H., and R. Bless, Ed., "Authorization for NSIS Signaling Layer Protocols", RFC 5981, DOI 10.17487/RFC5981, February 2011, <https://www.rfc-editor.org/info/rfc5981>.

[RFC6398] Le Faucheur, F., Ed., "IP Router Alert Considerations and Usage", BCP 168, RFC 6398, DOI 10.17487/RFC6398, October 2011, <https://www.rfc-editor.org/info/rfc6398>.

[RFC6928] Chu, J., Dukkipati, N., Cheng, Y., and M. Mathis, "Increasing TCP's Initial Window", RFC 6928, DOI 10.17487/RFC6928, April 2013, <https://www.rfc-editor.org/info/rfc6928>.

[RFC6951] Tuexen, M. and R. Stewart, "UDP Encapsulation of Stream Control Transmission Protocol (SCTP) Packets for End-Host to End-Host Communication", RFC 6951, DOI 10.17487/RFC6951, May 2013, <https://www.rfc-editor.org/info/rfc6951>.

[RFC7305] Lear, E., Ed., "Report from the IAB Workshop on Internet Technology Adoption and Transition (ITAT)", RFC 7305, DOI 10.17487/RFC7305, July 2014, <https://www.rfc-editor.org/info/rfc7305>.

[RFC7418] Dawkins, S., Ed., "An IRTF Primer for IETF Participants", RFC 7418, DOI 10.17487/RFC7418, December 2014, <https://www.rfc-editor.org/info/rfc7418>.

[RFC8170] Thaler, D., Ed., "Planning for Protocol Adoption and Subsequent Transitions", RFC 8170, DOI 10.17487/RFC8170, May 2017, <https://www.rfc-editor.org/info/rfc8170>.

[SAF07] Sarolahti, P., Allman, M., and S. Floyd, "Determining an appropriate sending rate over an underutilized network path", Computer Networking Volume 51, Number 7, May 2007.

[Sch11] Scharf, M., "Fast Startup Internet Congestion Control for Broadband Interactive Applications", Ph.D. Thesis, University of Stuttgart, April 2011.

[Shim6-35]
Meyer, D., Huston, G., Schiller, J., and V. Gill, "IAB IPv6 Multihoming Panel at NANOG 35", NANOG North American Network Operator Group, October 2005, <https://www.youtube.com/watch?v=ji6Y_rYHAQs>.

[TRIGTRAN-55]
"Triggers for Transport BOF at IETF 55", July 2003, <https://www.ietf.org/proceedings/55/239.htm>.

[TRIGTRAN-56]
"Triggers for Transport BOF at IETF 56", November 2003, <https://www.ietf.org/proceedings/56/251.htm>.

Author's Address

Spencer Dawkins (editor)
Huawei Technologies

Email: spencerdawkins.ietf@gmail.com

Path Aware Networking: Obstacles to Deployment (A Bestiary of Roads Not Taken)

draft-irtf-panrg-what-not-to-do-14

Abstract

At the first meeting of the Path Aware Networking Research Group, the research group agreed to catalog and analyze past efforts to develop and deploy Path Aware techniques, most of which were unsuccessful or at most partially successful, in order to extract insights and lessons for path-aware networking researchers.

This document contains that catalog and analysis.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document describes the lessons that IETF participants have learned (and learned the hard way) about Path Aware Networking over a period of several decades, and provides an analysis of reasons why various Path Aware Networking techniques have seen limited or no deployment.

1.1. What Does "Path Awareness" Mean in this Document?

The current definition of "Path Awareness", used by the Path Aware Networking Research Group, appears in Section 1.1 ("Definition") in [I-D.irtf-panrg-questions]. That definition is included here as a convenience to the reader.

> For purposes of this document, "path aware networking" describes endpoint discovery of the properties of paths they use for communication, and endpoint reaction to these properties that affects routing and/or transmission; note that this can and already does happen to some extent in the current Internet architecture. Expanding on this definition, a "path aware internetwork" is one in which endpoint discovery of path properties and endpoint selection of paths used by traffic exchanged by the endpoint are explicitly supported, regardless of the specific design of the protocol features which enable this discovery and selection.

Because this document reflects work performed over several decades, some technologies described in Section 5 may not reflect the current definition, but these technologies were considered "path aware" by their contributors, so these contributions are included in this retrospective document.

1.2. Note to RFC Editor

If the "Definition" in Section 1.1 ("Definition") of [I-D.irtf-panrg-questions] changes, the text in Section 1.1 of this document should be should be changed as well.

Whether that happens or not, the RFC Editor is requested to remove this section.

2. A Perspective On This Document

At the first meeting of the Path Aware Networking Research Group [PANRG], at IETF 99 [PANRG-99], Olivier Bonaventure led a discussion of "A Decade of Path Awareness" [PATH-Decade], on attempts, which were mostly unsuccessful for a variety of reasons, to exploit Path Aware techniques and achieve a variety of goals over the past decade. At the end of that discussion, two things were abundantly clear.

* The Internet community has accumulated considerable experience with many Path Aware techniques over a long period of time, and

* Although some path aware techniques have been deployed (for example, Differentiated Services, or DiffServ [RFC2475]), most of these techniques haven't seen widespread adoption and deplyment. Even "successful" techniques like DiffServ can face obstacles that prevents wider usage. The reasons for non-adoption and limited adoption and deployment are many, and are worthy of study.

The meta-lessons from that experience were

* Path aware networking has been more Research than Engineering, so establishing an IRTF Research Group for Path Aware Networking is the right thing to do [RFC7418].

* Analyzing a catalog of past experience to learn the reasons for non-adoption would be a great first step for the Research Group.

Allison Mankin, as IRTF Chair, officially chartered the Path Aware Networking Research Group in July, 2018.

This document contains the analysis performed by that research group (Section 4), based on that catalog (Section 5).

This document represents the consensus of the Path Aware Networking Research Group.

2.1. Notes for the Reader

This Informational document discusses Path Aware protocol mechanisms considered, and in some cases standardized, by the Internet Engineering Task Force (IETF), and considers Lessons Learned from those mechanisms. The intention is to inform the work of protocol designers, whether in the IRTF, the IETF, or elsewhere in the Internet ecosystem.

As an Informational document published in the IRTF stream, this document has no authority beyond the quality of the analysis it contains.

2.2. A Note About Path-Aware Techniques Included In This Document

This document does not catalog every proposed path aware networking technique that was not adopted and deployed. Instead, we limited our focus to technologies that passed through the IETF community, and still identified enough techniques to provide background for the lessons included in Section 4 to inform researchers and protocol engineers in their work.

No shame is intended for the techniques included in this document. As shown in Section 4, the quality of specific techniques had little to do with whether they were deployed or not. Based on the techniques cataloged in this document, it is likely that when these techniques were put forward, the proponents were trying to engineer something that could not be engineered without first carrying out research. Actual shame would be failing to learn from experience, and failing to share that experience with other networking researchers and engineers.

2.3. Venue for Discussion of this Document

(RFC Editor: please remove this section before publication)

Discussion of specific contributed experiences and this document in general should take place on the PANRG mailing list.

2.4. Architectural Guidance

As background for understanding the Lessons Learned contained in this document, the reader is encouraged to become familiar with the Internet Architecture Board's documents on "What Makes for a Successful Protocol?" [RFC5218] and "Planning for Protocol Adoption and Subsequent Transitions" [RFC8170].

Although these two documents do not specifically target path-aware networking protocols, they are helpful resources for readers seeking to improve their understanding of considerations for successful adoption and deployment of any protocol. For example, the Basic Success Factors described in Setion 2.1 of [RFC5218] are helpful for readers of this document.

Because there is an economic aspect to decisions about deployment, the IAB Workshop on Internet Technology Adoption and Transition [ITAT] report [RFC7305] also provides food for thought.

Several of the Lessons Learned in Section 4 reflect considerations described in [RFC5218], [RFC7305], and [RFC8170].

2.5. Terminology Used in this Document

The terms Node and Element in this document have the meaning defined in [PathProp].

2.6. Methodology for Contributions

This document grew out of contributions by various IETF participants with experience with one or more Path Aware Networking techniques.

There are many things that could be said about the Path Aware networking techniques that have been developed. For the purposes of this document, contributors were requested to provide

* the name of a technique, including an abbreviation if one was used

* if available, a long-term pointer to the best reference describing the technique

* a short description of the problem the technique was intended to solve

* a short description of the reasons why the technique wasn't adopted

* a short statement of the lessons that researchers can learn from our experience with this technique.

3. Applying the Lessons We've Learned

The initial scope for this document was roughly "what mistakes have we made in the decade prior to [PANRG-99], that we shouldn't make again". Some of the contributions in Section 5 predate the initial scope. The earliest Path-Aware Networking technique referred to in Section 5 is Section 5.1, published in the late 1970s. Given that the networking ecosystem has evolved continuously, it seems reasonable to consider how to apply these lessons.

The PANRG Research Group reviewed the Lessons Learned (Section 4) contained in the May 23, 2019 version of this document at IETF 105 [PANRG-105-Min], and carried out additional discussion at IETF 106 [PANRG-106-Min]. Table 1 provides the "sense of the room" about each lesson after those discussions. The intention is to capture whether a specific lesson seems to be

* "Invariant" - well-understood and is likely to be applicable for any proposed Path Aware Networking solution.

* "Variable" - has impeded deployment in the past, but might not be applicable in a specific technique. Engineering analysis to understand whether the lesson is applicable is prudent.

* "Not Now" - this characteristic tends to turn up a minefield full of dragons, and prudent network engineers will wish to avoid gambling on a technique that relies on this, until something significant changes

Lesson	Category
Justifying Deployment (Section 4.1)	Invariant
Providing Benefits for Early Adopters (Section 4.2)	Invariant
Providing Benefits during Partial Deployment (Section 4.3)	Invariant
Outperforming End-to-end Protocol Mechanisms (Section 4.4)	Variable
Paying for Path Aware Techniques (Section 4.5)	Invariant
Impact on Operational Practices (Section 4.6)	Invariant
Per-connection State (Section 4.7)	Variable
Keeping Traffic on Fast-paths (Section 4.8)	Variable
Endpoints Trusting Intermediate Nodes (Section 4.9)	Not Now
Intermediate Nodes Trusting Endpoints (Section 4.10)	Not Now
Reacting to Distant Signals (Section 4.11)	Variable
Support in Endpoint Protocol Stacks (Section 4.12)	Variable

Table 1

"Justifying Deployment", "Providing Benefits for Early Adopters", "Paying for Path Aware Techniques", and "Impact on Operational Practice" were considered to be invariant - the sense of the room was that these would always be considerations for any proposed Path Aware Technique.

"Providing Benefits During Partial Deployment" was added after IETF 105, during research group last call, and is also considered to be invariant.

For "Outperforming End-to-end Protocol Mechanisms", there is a trade-off between improved performance from Path Aware Techniques and additional complexity required by some Path Aware Techniques.

* For example, if you can obtain the same understanding of path characteristics from measurements obtained over a few more round trips, endpoint implementers are unlikely to be eager to add complexity, and many attributes can be measured from an endpoint, without assistance from intermediate nodes.

For "Per-connection State", the key questions discussed in the research group were "how much state" and "where state is maintained".

* IntServ (Section 5.2) required state at every intermediate node for every connection between two endpoints. As the Internet ecosystem has evolved, carrying many connections in a tunnel that appears to intermediate nodes as a single connection has become more common, so that additional end-to-end connections don't add additional state to intermediate nodes between tunnel endpoints. If these tunnels are encrypted, intermediate nodes between tunnel endpoints can't distinguish between connections, even if that were desirable.

For "Keeping Traffic on Fast-paths", we noted that this was true for many platforms, but not for all.

* For backbone routers, this is likely an invariant, but for platforms that rely more on general-purpose computers to make forwarding decisions, this may not be a fatal flaw for Path Aware Networking techniques.

For "Endpoints Trusting Intermediate Nodes" and "Intermediate Nodes Trusting Endpoints", these lessons point to the broader need to revisit the Internet Threat Model.

* We noted with relief that discussions about this were already underway in the IETF community at IETF 105 (see the Security Area Open Meeting minutes [SAAG-105-Min] for discussion of [draft-arkko-arch-internet-threat-model] and [draft-farrell-etm]), and the Internet Architecture Board has created a mailing list for continued discussions ([model-t]), but we recognize that there are Path Aware Networking aspects of this effort, requiring research.

For "Reacting to Distant Signals", we noted that not all attributes are equal.

* If an attribute is stable over an extended period of time, is difficult to observe via end-to-end mechanisms, and is valuable, Path Aware Techniques that rely on that attribute to provide a significant benefit become more attractive.

* Analysis to help identify attributes that are useful enough to justify deployment of Path Aware techniques that make use of those attributes would be helpful.

For "Support in Endpoint Protocol Stacks", we noted that Path Aware applications must be able to identify and communicate requirements about path characteristics.

* The de-facto sockets API has no way of signaling application expectations for the network path to the protocol stack.

4. Summary of Lessons Learned

This section summarizes the Lessons Learned from the contributed subsections in Section 5.

Each Lesson Learned is tagged with one or more contributions that encountered this obstacle as a significant impediment to deployment. Other contributed techniques may have also encountered this obstacle, but this obstacle may not have been the biggest impediment to deployment for those techniques.

It is useful to notice that sometimes an obstacle might impede deployment, while at other times, the same obstacle might prevent adoption and deployment entirely. The research group discussed distinguishing between obstacles that impede and obstacles that prevent, but it appears that the boundary between "impede" and "prevent" can shift over time - some of the Lessons Learned are based on both Path Aware techniques that were not deployed, and Path Aware techniques that were deployed, but were not deployed widely or quickly. See Section 5.6 and Section 5.6.3 as one example of this shifting boundary.

4.1. Justifying Deployment

The benefit of Path Awareness must be great enough to justify making changes in an operational network. The colloquial U.S. American English expression, "If it ain't broke, don't fix it" is a "best current practice" on today's Internet. (See Section 5.3, Section 5.5, and Section 5.4, in addition to [RFC5218]).

4.2. Providing Benefits for Early Adopters

Providing benefits for early adopters can be key - if everyone must deploy a technique in order for the technique to provide benefits, or even to work at all, the technique is unlikely to be adopted widely or quickly. (See Section 5.2 and Section 5.3, in addition to [RFC5218]).

4.3. Providing Benefits During Partial Deployment

Some proposals require that all path elements along the full length of the path must be upgraded to support a new technique, before any benefits can be seen. This is likely to require coordination between operators who control a subset of path elements, and between operators and end users if endpoint upgrades are required. If a technique provides benefits when only a part of the path has been upgraded, this is likely to encourage adoption and deployment. (See Section 5.2 and Section 5.3, in addition to [RFC5218]).

4.4. Outperforming End-to-end Protocol Mechanisms

Adaptive end-to-end protocol mechanisms may respond to feedback quickly enough that the additional realizable benefit from a new Path Aware mechanism that tries to manipulate nodes along a path, or observe the attributes of nodes along a path, may be much smaller than anticipated (Section 5.3 and Section 5.5).

4.5. Paying for Path Aware Techniques

"Follow the money." If operators can't charge for a Path Aware technique to recover the costs of deploying it, the benefits to the operator must be really significant. Corollary: If operators charge for a Path Aware technique, the benefits to users of that Path Aware technique must be significant enough to justify the cost. (See Section 5.1, Section 5.2, and Section 5.5).

4.6. Impact on Operational Practices

Impact of a Path Aware technique requiring changes to operational practices can affect how quickly or widely a promising technique is deployed. The impacts of these changes may make deployment more likely, but often discourage deployment. (See Section 5.6, including Section 5.6.3).

4.7. Per-connection State

Per-connection state in intermediate nodes has been an impediment to adoption and deployment in the past, because of added cost and complexity. Often, similar benefits can be achieved with much less finely-grained state. This is especially true as we move from the edge of the network, further into the routing core (See Section 5.1 and Section 5.2).

4.8. Keeping Traffic on Fast-paths

Many modern platforms, especially high-end routers, have been designed with hardware that can make simple per-packet forwarding decisions ("fast-paths"), but have not been designed to make heavy use of in-band mechanisms such as IPv4 and IPv6 Router Alert Options (RAO) that require more processing to make forwarding decisions. Packets carrying in-band mechanisms are diverted to other processors in the router with much lower packet processing rates. Operators can be reluctant to deploy techniques that rely heavily on in-band mechanisms because they may significantly reduce packet throughput. (See Section 5.7).

4.9. Endpoints Trusting Intermediate Nodes

If intermediate nodes along the path can't be trusted, it's unlikely that endpoints will rely on signals from intermediate nodes to drive changes to endpoint behaviors. (See Section 5.5, Section 5.4). We note that "trust" is not binary - one, low, level of trust applies when a node issuing a message can confirm that it has visibility of the packets on the path it is seeking to control [RFC8085] (e.g., an ICMP message included a quoted packet from the source). A higher level of trust can arise when an endpoint has established a short term, or even long term, trust relationship with network nodes.

4.10. Intermediate Nodes Trusting Endpoints

If the endpoints do not have any trust relationship with the intermediate nodes along a path, operators have been reluctant to deploy techniques that rely on endpoints sending unauthenticated control signals to routers. (See Section 5.2 and Section 5.7. We also note this still remains a factor hindering deployment of DiffServ).

4.11. Reacting to Distant Signals

Because the Internet is a distributed system, if the distance that information from distant path elements travels to a Path Aware host is sufficiently large, the information may no longer accurately represent the state and situation at the distant host or elements along the path when it is received locally. In this case, the benefit that a Path Aware technique provides will be inconsistent, and may not always be beneficial. (See Section 5.3).

4.12. Support in Endpoint Protocol Stacks

Just because a protocol stack provides a new feature/signal does not mean that applications will use the feature/signal. Protocol stacks may not know how to effectively utilize Path-Aware techniques, because the protocol stack may require information from applications to permit the technique to work effectively, but applications may not a-priori know that information. Even if the application does know that information, the de-facto sockets API has no way of signaling application expectations for the network path to the protocol stack. In order for applications to provide these expectations to protocol stacks, we need an API that signals more than the packets to be sent. (See Section 5.1 and Section 5.2).

5. Contributions

Contributions on these Path Aware networking techniques were analyzed to arrive at the Lessons Learned captured in Section 4.

Our expectation is that most readers will not need to read through this section carefully, but we wanted to record these hard-fought lessons as a service to others who may revisit this document, so they'll have the details close at hand.

5.1. Stream Transport (ST, ST2, ST2+)

The suggested references for Stream Transport are:

* ST - A Proposed Internet Stream Protocol [IEN-119]

* Experimental Internet Stream Protocol, Version 2 (ST-II) [RFC1190]

* Internet Stream Protocol Version 2 (ST2) Protocol Specification - Version ST2+ [RFC1819]

The first version of Stream Transport, ST [IEN-119], was published in the late 1970's and was implemented and deployed on the ARPANET at small scale. It was used throughout the 1980's for experimental transmission of voice, video, and distributed simulation.

The second version of the ST specification (ST2) [RFC1190] [RFC1819] was an experimental connection-oriented internetworking protocol that operated at the same layer as connectionless IP. ST2 packets could be distinguished by their IP header protocol numbers (IP, at that time, used protocol number 4, while ST2 used protocol number 5).

ST2 used a control plane layered over IP to select routes and reserve capacity for real-time streams across a network path, based on a flow specification communicated by a separate protocol. The flow specification could be associated with QoS state in routers, producing an experimental resource reservation protocol. This allowed ST2 routers along a path to offer end-to-end guarantees, primarily to satisfy the QoS requirements for realtime services over the Internet.

5.1.1. Reasons for Non-deployment

Although implemented in a range of equipment, ST2 was not widely used after completion of the experiments. It did not offer the scalability and fate-sharing properties that have come to be desired by the Internet community.

The ST2 protocol is no longer in use.

5.1.2. Lessons Learned.

As time passed, the trade-off between router processing and link capacity changed. Links became faster and the cost of router processing became comparatively more expensive.

The ST2 control protocol used "hard state" - once a route was established, and resources were reserved, routes and resources existing until they were explicitly released via signaling. A soft-state approach was thought superior to this hard-state approach, and led to development of the IntServ model described in Section 5.2.

5.2. Integrated Services (IntServ)

The suggested references for IntServ are:

* RFC 1633 Integrated Services in the Internet Architecture: an Overview [RFC1633]

* RFC 2211 Specification of the Controlled-Load Network Element Service [RFC2211]

* RFC 2212 Specification of Guaranteed Quality of Service [RFC2212]

* RFC 2215 General Characterization Parameters for Integrated Service Network Elements [RFC2215]

* RFC 2205 Resource ReSerVation Protocol (RSVP) [RFC2205]

In 1994, when the IntServ architecture document [RFC1633] was published, real-time traffic was first appearing on the Internet. At that time, bandwidth was still a scarce commodity. Internet Service Providers built networks over DS3 (45 Mbps) infrastructure, and sub-rate (< 1 Mpbs) access was common. Therefore, the IETF anticipated a need for a fine-grained QoS mechanism.

In the IntServ architecture, some applications can require service guarantees. Therefore, those applications use the Resource Reservation Protocol (RSVP) [RFC2205] to signal QoS reservations across network paths. Every router in the network maintains per-flow soft-state to a) perform call admission control and b) deliver guaranteed service.

Applications use Flow Specification (Flow Specs) [RFC2210] to describe the traffic that they emit. RSVP reserves capacity for traffic on a per Flow Spec basis.

5.2.1. Reasons for Non-deployment

Although IntServ has been used in enterprise and government networks, IntServ was never widely deployed on the Internet because of its cost. The following factors contributed to operational cost:

* IntServ must be deployed on every router that is on a path where IntServ is to be used

* IntServ maintained per flow state

As IntServ was being discussed, the following occurred:

* For many expected uses, it became more cost effective to solve the QoS problem by adding bandwidth. Between 1994 and 2000, Internet Service Providers upgraded their infrastructures from DS3 (45 Mbps) to OC-48 (2.4 Gbps). This meant that even if an endpoint was using IntServ in an IntServ-enabled network, its requests would never be denied, so endpoints and Internet Service Providers had little reason to enable IntServ.

* DiffServ [RFC2475] offered a more cost-effective, albeit less fine-grained, solution to the QoS problem.

5.2.2. Lessons Learned.

The following lessons were learned:

* Any mechanism that requires every onpath router to maintain per-flow state is not likely to succeed, unless the additional cost for offering the feature can be recovered from the user.

* Any mechanism that requires an operator to upgrade all of its routers is not likely to succeed, unless the additional cost for offering the feature can be recovered from the user.

In environments where IntServ has been deployed, trust relationships with endpoints are very different from trust relationships on the Internet itself, and there are often clearly-defined hierarchies in Service Level Agreements (SLAs), and well-defined transport flows operating with pre-determined capacity and latency requirements over paths where capacity or other attributes are constrained.

IntServ was never widely deployed to manage capacity across the Internet. However, the technique that it produced was deployed for reasons other than bandwidth management. RSVP is widely deployed as an MPLS signaling mechanism. BGP reuses the RSVP concept of Filter Specs to distribute firewall filters, although they are called Flow Spec Component Types in BGP [RFC5575].

5.3. Quick-Start TCP

The suggested references for Quick-Start TCP are:

* Quick-Start for TCP and IP [RFC4782]

* Determining an appropriate initial sending rate over an underutilized network path [SAF07]

* Fast Startup Internet Congestion Control for Broadband Interactive Applications [Sch11]

* Using Quick-Start to enhance TCP-friendly rate control performance in bidirectional satellite networks [QS-SAT]

Quick-Start [RFC4782] is an Experimental TCP extension that leverages support from the routers on the path to determine an allowed initial sending rate for a path through the Internet, either at the start of data transfers or after idle periods. Without information about the path, a sender cannot easily determine an appropriate initial sending rate. The default TCP congestion control therefore uses the safe but time-consuming slow-start algorithm [RFC5681]. With Quick-Start, connections are allowed to use higher initial sending rates if there is significant unused bandwidth along the path, and if the sender and all of the routers along the path approve the request.

By examining the Time To Live (TTL) field in Quick-Start packets, a sender can determine if routers on the path have approved the Quick-Start request. However, this method is unable to take into account the routers hidden by tunnels or other network nodes invisible at the IP layer.

The protocol also includes a nonce that provides protection against cheating routers and receivers. If the Quick-Start request is explicitly approved by all routers along the path, the TCP host can send at up to the approved rate; otherwise TCP would use the default congestion control. Quick-Start requires modifications in the involved end-systems as well in routers. Due to the resulting deployment challenges, Quick-Start was only proposed in [RFC4782] for controlled environments.

The Quick-Start mechanism is a lightweight, coarse-grained, in-band, network-assisted fast startup mechanism. The benefits are studied by simulation in a research paper [SAF07] that complements the protocol specification. The study confirms that Quick-Start can significantly speed up mid-sized data transfers. That paper also presents router algorithms that do not require keeping per-flow state. Later studies [Sch11] comprehensively analyzes Quick-Start with a full Linux implementation and with a router fast path prototype using a network processor. In both cases, Quick-Start could be implemented with limited additional complexity.

5.3.1. Reasons for Non-deployment

However, experiments with Quick-Start in [Sch11] revealed several challenges:

* Having information from the routers along the path can reduce the risk of congestion, but cannot avoid it entirely. Determining whether there is unused capacity is not trivial in actual router and host implementations. Data about available capacity visible at the IP layer may be imprecise, and due to the propagation delay, information can already be outdated when it reaches a sender. There is a trade-off between the speedup of data transfers and the risk of congestion even with Quick-Start. This could be mitigated by only allowing Quick-Start to access a proportion of the unused capacity along a path.

* For scalable router fast path implementation, it is important to enable parallel processing of packets, as this is a widely used method e.g. in network processors. One challenge is synchronization of information between packets that are processed in parallel, which should be avoided as much as possible.

* Only some types of application traffic can benefit from Quick-Start. Capacity needs to be requested and discovered. The discovered capacity needs to be utilized by the flow, or it implicitly becomes available for other flows. Failing to use the requested capacity may have already reduced the pool of Quick-Start capacity that was made available to other competing Quick-Start requests. The benefit is greatest when senders use this only for bulk flows and avoid sending unnecessary Quick-Start requests, e.g. for flows that only send a small amount of data. Choosing an appropriate request size requires application-internal knowledge that is not commonly expressed by the transport API. How a sender can determine the rate for an initial Quick-Start request is still a largely unsolved problem.

There is no known deployment of Quick-Start for TCP or other IETF transports.

5.3.2. Lessons Learned

Some lessons can be learned from Quick-Start. Despite being a very light-weight protocol, Quick-Start suffers from poor incremental deployment properties, both regarding the required modifications in network infrastructure as well as its interactions with applications. Except for corner cases, congestion control can be quite efficiently performed end-to-end in the Internet, and in modern stacks there is not much room for significant improvement by additional network support.

After publication of the Quick-Start specification, there have been large-scale experiments with an initial window of up to 10 MSS [RFC6928]. This alternative "IW10" approach can also ramp-up data transfers faster than the standard congestion control, but it only requires sender-side modifications. As a result, this approach can be easier and incrementally deployed in the Internet. While theoretically Quick-Start can outperform "IW10", the improvement in completion time for data transfer times can, in many cases, be small. After publication of [RFC6928], most modern TCP stacks have increased their default initial window.

5.4. ICMP Source Quench

The suggested references for ICMP Source Quench are:

* INTERNET CONTROL MESSAGE PROTOCOL [RFC0792]

The ICMP Source Quench message [RFC0792] allowed an on-path router to request the source of a flow to reduce its sending rate. This method allowed a router to provide an early indication of impending congestion on a path to the sources that contribute to that congestion.

5.4.1. Reasons for Non-deployment

This method was deployed in Internet routers over a period of time, the reaction of endpoints to receiving this signal has varied. For low speed links, with low multiplexing of flows the method could be used to regulate (momentarily reduce) the transmission rate. However, the simple signal does not scale with link speed, or the number of flows sharing a link.

The approach was overtaken by the evolution of congestion control methods in TCP [RFC2001], and later also by other IETF transports. Because these methods were based upon measurement of the end-to-end path and an algorithm in the endpoint, they were able to evolve and mature more rapidly than methods relying on interactions between operational routers and endpoint stacks.

After ICMP Source Quench was specified, the IETF began to recommend that transports provide end-to-end congestion control [RFC2001]. The Source Quench method has been obsoleted by the IETF [RFC6633], and both hosts and routers must now silently discard this message.

5.4.2. Lessons Learned

This method had several problems:

First, [RFC0792] did not sufficiently specify how the sender would react to the ICMP Source Quench signal from the path (e.g., [RFC1016]). There was ambiguity in how the sender should utilize this additional information. This could lead to unfairness in the way that receivers (or routers) responded to this message.

Second, while the message did provide additional information, the Explicit Congestion Notification (ECN) mechanism [RFC3168] provided a more robust and informative signal for network nodes to provide early indication that a path has become congested.

The mechanism originated at a time when the Internet trust model was very different. Most endpoint implementations did not attempt to verify that the message originated from an on-path node before they utilized the message. This made it vulnerable to denial of service attacks. In theory, routers might have chosen to use the quoted packet contained in the ICMP payload to validate that the message originated from an on-path node, but this would have increased per-packet processing overhead for each router along the path, would have required transport functionality in the router to verify whether the quoted packet header corresponded to a packet the router had sent. In addition, section 5.2 of [RFC4443] noted ICMPv6-based attacks on hosts that would also have threatened routers processing ICMPv6 Source Quench payloads. As time passed, it became increasingly obvious that the lack of validation of the messages exposed receivers to a security vulnerability where the messages could be forged to create a tangible denial of service opportunity.

5.5. Triggers for Transport (TRIGTRAN)

The suggested references for TRIGTRAN are:

* TRIGTRAN BOF at IETF 55 [TRIGTRAN-55]

* TRIGTRAN BOF at IETF 56 [TRIGTRAN-56]

TCP [RFC0793] has a well-known weakness - the end-to-end flow control mechanism has only a single signal, the loss of a segment, and TCP implementations since the late 1980s have interpreted the loss of a segment as evidence that the path between two endpoints may have become congested enough to exhaust buffers on intermediate hops, so that the TCP sender should "back off" - reduce its sending rate until it knows that its segments are now being delivered without loss [RFC5681]. More modern TCP stacks have added a growing array of strategies about how to establish the sending rate [RFC5681], but when a path is no longer operational, TCP would continue to retry transmissions, which would fail, again, and double their Retransmission Time Out (RTO) timers with each failed transmission, with the result that TCP would wait many seconds before retrying a segment, even if the path becomes operational while the sender is waiting for its next retry.

The thinking behind TRIGTRAN was that if a path completely stopped working because a link along the path was "down", somehow something along the path could signal TCP when that link returned to service, and the sending TCP could retry immediately, without waiting for a full retransmission timeout (RTO) period.

5.5.1. Reasons for Non-deployment

The early dreams for TRIGTRAN were dashed because of an assumption that TRIGTRAN triggers would be unauthenticated. This meant that any "safe" TRIGTRAN mechanism would have relied on a mechanism such as setting the IPv4 TTL or IPv6 Hop Count to 255 at a sender and testing that it was 254 upon receipt, so that a receiver could verify that a signal was generated by an adjacent sender known to be on the path being used, and not some unknown sender which might not even be on the path (e.g., "The Generalized TTL Security Mechanism (GTSM)" [RFC5082]). This situation is very similar to the case for ICMP Source Quench messages as described in Section 5.4, which were also unauthenticated, and could be sent by an off-path attacker, resulting in deprecation of ICMP Source Quench message processing [RFC6633].

TRIGTRAN's scope shrunk from "the path is down" to "the first-hop link is down".

But things got worse.

Because TRIGTRAN triggers would only be provided when the first-hop link was "down", TRIGTRAN triggers couldn't replace normal TCP retransmission behavior if the path failed because some link further along the network path was "down". So TRIGTRAN triggers added complexity to an already complex TCP state machine, and did not allow any existing complexity to be removed.

There was also an issue that the TRIGTRAN signal was not sent in response to a specific host that had been sending packets, and was instead a signal that stimulated a response by any sender on the link. This needs to scale when there are multiple flows trying to use the same resource, yet the sender of a trigger has no understanding how many of the potential traffic sources will respond by sending packets - if recipients of the signal back-off their responses to a trigger to improve scaling, then that immediately mitigates the benefit of the signal.

Finally, intermediate forwarding nodes required modification to provide TRIGTRAN triggers, but operators couldn't charge for TRIGTRAN triggers, so there was no way to recover the cost of modifying, testing, and deploying updated intermediate nodes.

Two TRIGTRAN BOFs were held, at IETF 55 [TRIGTRAN-55] and IETF 56 [TRIGTRAN-56], but this work was not chartered, and there was no interest in deploying TRIGTRAN unless it was chartered and standardized in the IETF.

5.5.2. Lessons Learned.

The reasons why this work was not chartered, much less deployed, provide several useful lessons for researchers.

* TRIGTRAN started with a plausible value proposition, but networking realities in the early 2000s forced reductions in scope that led directly to reductions in potential benefits, but no corresponding reductions in costs and complexity.

* These reductions in scope were the direct result of an inability for hosts to trust or authenticate TRIGTRAN signals they received from the network.

* Operators did not believe they could charge for TRIGTRAN signaling, because first-hop links didn't fail frequently, and TRIGTRAN provided no reduction in operating expenses, so there was little incentive to purchase and deploy TRIGTRAN-capable network equipment.

It is also worth noting that the targeted environment for TRIGTRAN in the late 1990s contained links with a relatively small number of directly-connected hosts - for instance, cellular or satellite links. The transport community was well aware of the dangers of sender synchronization based on multiple senders receiving the same stimulus at the same time, but the working assumption for TRIGTRAN was that there wouldn't be enough senders for this to be a meaningful problem. In the 2010s, it is common for a single "link" to support many senders and receivers on a single link, likely requiring TRIGTRAN senders to wait some random amount of time before sending after receiving a TRIGTRAN signal, which would have reduced the benefits of TRIGTRAN even more.

5.6. Shim6

The suggested references for Shim6 are:

* Shim6: Level 3 Multihoming Shim Protocol for IPv6 [RFC5533]

The IPv6 routing architecture [RFC1887] assumed that most sites on the Internet would be identified by Provider Assigned IPv6 prefixes, so that Default-Free Zone routers only contained routes to other providers, resulting in a very small IPv6 global routing table.

For a single-homed site, this could work well. A multihomed site with only one upstream provider could also work well, although BGP multihoming from a single upstream provider was often a premium service (costing more than twice as much as two single-homed sites), and if the single upstream provider went out of service, all of the multihomed paths could fail simultaneously.

IPv4 sites often multihomed by obtaining Provider Independent prefixes, and advertising these prefixes through multiple upstream providers. With the assumption that any multihomed IPv4 site would also multihome in IPv6, it seemed likely that IPv6 routing would be subject to the same pressures to announce Provider Independent prefixes, resulting in a global IPv6 routing table that exhibited the same explosive growth as the global IPv4 routing table. During the early 2000s, work began on a protocol that would provide multihoming for IPv6 sites without requiring sites to advertise Provider Independent prefixes into the IPv6 global routing table.

This protocol, called Shim6, allowed two endpoints to exchange multiple addresses ("Locators") that all mapped to the same endpoint ("Identity"). After an endpoint learned multiple Locators for the other endpoint, it could send to any of those Locators with the expectation that those packets would all be delivered to the endpoint with the same Identity. Shim6 was an example of an "Identity/Locator Split" protocol.

Shim6, as defined in [RFC5533] and related RFCs, provided a workable solution for IPv6 multihoming using Provider Assigned prefixes, including capability discovery and negotiation, and allowing end-to-end application communication to continue even in the face of path failure, because applications don’t see Locator failures, and continue to communicate with the same Identity using a different Locator.

5.6.1. Reasons for Non-deployment

Note that the problem being addressed was "site multihoming", but Shim6 was providing "host multihoming". That meant that the decision about what path would be used was under host control, not under router control.

Although more work could have been done to provide a better technical solution, the biggest impediments to Shim6 deployment were operational and business considerations. These impediments were discussed at multiple network operator group meetings, including [Shim6-35] at [NANOG-35].

The technique issues centered around concerns that Shim6 relied on the host to track all the connections, while also tracking Identity/ Locator mappings in the kernel, and tracking failures to recognize that a backup path has failed.

The operator issues centered around concerns that operators were performing traffic engineering, but would have no visibility or control over hosts when they chose to begin using another path, and relying on hosts to engineer traffic exposed their networks to oscillation based on feedback loops, as hosts move from path to path. At a minimum, traffic engineering policies must be pushed down to individual hosts. In addition, firewalls that expected to find a transport-level protocol header in the IP payload, would see a Shim6 Identity header, and be unable to perform transport-protocol-based firewalling functions because its normal processing logic would not look past the Identity header.

The business issues centered removing or reducing the ability to sell BGP multihoming service, which is often more expensive than single-homed connectivity.

5.6.2. Lessons Learned

It is extremely important to take operational concerns into account when a path-aware protocol is making decisions about path selection that may conflict with existing operational practices and business considerations.

5.6.3. Addendum on MultiPath TCP

During discussions in the PANRG session at IETF 103 [PANRG-103-Min], Lars Eggert, past Transport Area Director, pointed out that during charter discussions for the Multipath TCP working group [MP-TCP], operators expressed concerns that customers could use Multipath TCP to loadshare TCP connections across operators simultaneously and compare passive performance measurements across network paths in real time, changing the balance of power in those business relationships. Although the Multipath TCP working group was chartered, this concern could have acted as an obstacle to deployment.

Operator objections to Shim6 were focused on technical concerns, but this concern could have also been an obstacle to Shim6 deployment if the technical concerns had been overcome.

5.7. Next Steps in Signaling (NSIS)

The suggested references for Next Steps in Signaling (NSIS) are:

* the concluded working group charter [NSIS-CHARTER-2001]

* GIST: General Internet Signalling Transport [RFC5971]

* NAT/Firewall NSIS Signaling Layer Protocol (NSLP) [RFC5973]

* NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling [RFC5974]

* Authorization for NSIS Signaling Layer Protocols [RFC5981]

The NSIS Working Group worked on signaling techniques for network layer resources (e.g., QoS resource reservations, Firewall and NAT traversal).

When RSVP [RFC2205] was used in deployments, a number of questions came up about its perceived limitations and potential missing features. The issues noted in the NSIS Working Group charter [NSIS-CHARTER-2001] include interworking between domains with different QoS architectures, mobility and roaming for IP interfaces, and complexity. Later, the lack of security in RSVP was also recognized ([RFC4094]).

The NSIS Working Group was chartered to tackle those issues and initially focused on QoS signaling as its primary use case. However, over time a new approach evolved that introduced a modular architecture using application-specific signaling protocols (the NSIS Signaling Layer Protocol (NSLP)) on top of a generic signaling transport protocol (the NSIS Transport Layer Protocol (NTLP)).

The NTLP is defined in [RFC5971]. Two NSLPs are defined: the NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling [RFC5974] as well as the NAT/Firewall NSIS Signaling Layer Protocol (NSLP) [RFC5973].

5.7.1. Reasons for Non-deployment

The obstacles for deployment can be grouped into implementation-related aspects and operational aspects.

* Implementation-related aspects:

Although NSIS provides benefits with respect to flexibility, mobility, and security compared to other network signaling techniques, hardware vendors were reluctant to deploy this solution, because it would require additional implementation effort and would result in additional complexity for router implementations.

The NTLP mainly operates as path-coupled signaling protocol, i.e, its messages are processed at the intermediate node's control plane that are also forwarding the data flows. This requires a mechanism to intercept signaling packets while they are forwarded in the same manner (especially along the same path) as data packets. NSIS uses the IPv4 and IPv6 Router Alert Option (RAO) to allow for interception of those path-coupled signaling messages, and this technique requires router implementations to correctly understand and implement the handling of RAOs, e.g., to only process packet with RAOs of interest and to leave packets with irrelevant RAOs in the fast forwarding processing path (a comprehensive discussion of these issues can be found in [RFC6398]). The latter was an issue with some router implementations at the time of standardization.

Another reason is that path-coupled signaling protocols that interact with routers and request manipulation of state at these routers (or any other network element in general) are under scrutiny: a packet (or sequence of packets) out of the mainly untrusted data path is requesting creation and manipulation of network state. This is seen as potentially dangerous (e.g., opens up a Denial of Service (DoS) threat to a router's control plane) and difficult for an operator to control. Path-coupled signaling approaches were considered problematic (see also section 3 of [RFC6398]). There are recommendations on how to secure NSIS nodes and deployments (e.g., [RFC5981]).

* Operational Aspects:

NSIS not only required trust between customers and their provider, but also among different providers. Especially, QoS signaling techniques would require some kind of dynamic service level agreement support that would imply (potentially quite complex) bilateral negotiations between different Internet service providers. This complexity was not considered to be justified and increasing the bandwidth (and thus avoiding bottlenecks) was cheaper than actively managing network resource bottlenecks by using path-coupled QoS signaling techniques. Furthermore, an end-to-end path typically involves several provider domains and these providers need to closely cooperate in cases of failures.

5.7.2. Lessons Learned

One goal of NSIS was to decrease the complexity of the signaling protocol, but a path-coupled signaling protocol comes with the intrinsic complexity of IP-based networks, beyond the complexity of the signaling protocol itself. Sources of intrinsic complexity include:

* the presence of asymmetric routes between endpoints and routers

* the lack of security and trust at large in the Internet infrastructure

* the presence of different trust boundaries

* the effects of best-effort networks (e.g., robustness to packet loss)

* divergence from the fate sharing principle (e.g., state within the network).

Any path-coupled signaling protocol has to deal with these realities.

Operators view the use of IPv4 and IPv6 Router Alert Option (RAO) to signal routers along the path from end systems with suspicion, because these end systems are usually not authenticated and heavy use of RAOs can easily increase the CPU load on routers that are designed to process most packets using a hardware "fast path" and diverting packets containing RAO to a slower, more capable processor.

5.8. IPv6 Flow Label

The suggested references for IPv6 Flow Label are:

* IPv6 Flow Label Specification [RFC6437]

IPv6 specifies a 20-bit field Flow Label field [RFC6437], included in the fixed part of the IPv6 header and hence present in every IPv6 packet. An endpoint sets the value in this field to one of a set of pseudo-randomly assigned values. If a packet is not part of any flow, the flow label value is set to zero [RFC3697]. A number of Standards Track and Best Current Practice RFCs (e.g., [RFC8085], [RFC6437], [RFC6438]) encourage IPv6 endpoints to set a non-zero value in this field. A multiplexing transport could choose to use multiple flow labels to allow the network to independently forward its subflows, or to use one common value for the traffic aggregate. The flow label is present in all fragments. IPsec was originally put forward as one important use-case for this mechanism and does encrypt the field [RFC6438].

Once set, the flow label can provide information that can help inform network nodes about subflows present at the transport layer, without needing to interpret the setting of upper layer protocol fields [RFC6294]. This information can also be used to coordinate how aggregates of transport subflows are grouped when queued in the network and to select appropriate per-flow forwarding when choosing between alternate paths [RFC6438] (e.g. for Equal Cost Multipath Routing (ECMP) and Link Aggregation (LAG)).

5.8.1. Reasons for Non-deployment

Despite the field being present in every IPv6 packet, the mechanism did not receive as much use as originally envisioned. One reason is that to be useful it requires engagement by two different stakeholders:

* Endpoint Implementation:

For network nodes along a path to utilize the flow label there needs to be a non-zero value inserted in the field [RFC6437] at the sending endpoint. There needs to be an incentive for an endpoint to set an appropriate non-zero value. The value should appropriately reflect the level of aggregation the traffic expects to be provided by the network. However, this requires the stack to know granularity at which flows should be identified (or conversely which flows should receive aggregated treatment), i.e., which packets carry the same flow label. Therefore, setting a non-zero value may result in additional choices that need to be made by an application developer.

Although the standard [RFC3697] forbids any encoding of meaning into the flow label value, the opportunity to use the flow label as a covert channel or to signal other meta-information may have raised concerns about setting a non-zero value [RFC6437].

Before methods are widely deployed to use this method, there could be no incentive for an endpoint to set the field.

* Operational support in network nodes:

A benefit can only be realized when a network node along the path also uses this information to inform its decisions. Network equipment (routers and/or middleboxes) need to include appropriate support so they can utilize the field when making decisions about how to classify flows, or to inform forwarding choices. Use of any optional feature in a network node also requires corresponding updates to operational procedures, and therefore is normally only introduced when the cost can be justified.

A benefit from utilizing the flow label is expected to be increased quality of experience for applications - but this comes at some operational cost to an operator, and requires endpoints to set the field.

5.8.2. Lessons Learned

The flow label is a general purpose header field for use by the path. Multiple uses have been proposed. One candidate use was to reduce the complexity of forwarding decisions. However, modern routers can use a "fast path", often taking advantage of hardware to accelerate processing. The method can assist in more complex forwarding, such as ECMP and load balancing.

Although [RFC6437] recommended that endpoints should by default choose uniformly-distributed labels for their traffic, the specification permitted an endpoint to choose to set a zero value. This ability of endpoints to choose to set a flow label of zero has had consequences on deployability:

* Before wide-scale support by endpoints, it would be impossible to rely on a non-zero flow label being set. Network nodes therefore would need to also employ other techniques to realize equivalent functions. An example of a method is one assuming semantics of the source port field to provide entropy input to a network-layer hash. This use of a 5-tuple to classify a packet represents a layering violation [RFC6294]. When other methods have been deployed, they increase the cost of deploying standards-based methods, even though they may offer less control to endpoints and result in potential interaction with other uses/interpretation of the field.

* Even though the flow label is specified as an end-to-end field, some network paths have been observed to not transparently forward the flow label. This could result from non-conformant equipment, or could indicate that some operational networks have chosen to re-use the protocol field for other (e.g. internal purposes). This results in lack of transparency, and a deployment hurdle to endpoints expecting that they can set a flow label that is utilized by the network. The more recent practice of "greasing" [GREASE] would suggest that a different outcome could have been achieved if endpoints were always required to set a non-zero value.

* [RFC1809] noted that setting the choice of the flow label value can depend on the expectations of the traffic generated by an application, which suggests an API should be presented to control the setting or policy that is used. However, many currently available APIs do not have this support.

A growth in the use of encrypted transports, (e.g. QUIC [QUIC-WG]) seems likely to raise similar issues to those discussed above and could motivate renewed interest in utilizing the flow label.

6. Security Considerations

This document describes Path Aware techniques that were not adopted and widely deployed on the Internet, so it doesn’t affect the security of the Internet.

If this document meets its goals, we may develop new techniques for Path Aware Networking that would affect the security of the Internet, but security considerations for those techniques will be described in the corresponding RFCs that specify them.

7. IANA Considerations

This document makes no requests of IANA.

8. Acknowledgments

Initial material for Section 5.1 on ST2 was provided by Gorry Fairhurst.

Initial material for Section 5.2 on IntServ was provided by Ron Bonica.

Initial material for Section 5.3 on Quick-Start TCP was provided by Michael Scharf, who also provided suggestions to improve this section after it was edited.

Initial material for Section 5.4 on ICMP Source Quench was provided by Gorry Fairhurst.

Initial material for Section 5.5 on Triggers for Transport (TRIGTRAN) was provided by Spencer Dawkins.

Section 5.6 on Shim6 builds on initial material describing obstacles provided by Erik Nordmark, with background added by Spencer Dawkins.

Initial material for Section 5.7 on Next Steps In Signaling (NSIS) was provided by Roland Bless and Martin Stiemerling.

Initial material for Section 5.8 on IPv6 Flow Labels was provided by Gorry Fairhurst.

Our thanks to C.M. Heard, David Black, Gorry Fairhurst, Joe Touch, Joeri de Ruiter, Mohamed Boucadair, Roland Bless, Ruediger Geib, Theresa Enghardt, and Wes Eddy, who provided review comments on previous versions.

Mallory Knodel reviewed this document for the Internet Engineering Steering Group, and provided many helpful suggestions.

Special thanks to Adrian Farrel for helping Spencer navigate the twisty little passages of Flow Specs and Filter Specs in IntServ, RSVP, MPLS, and BGP. They are all alike, except when they are different [Colossal-Cave].

9. Informative References

[Colossal-Cave]
"Wikipedia Page for Colossal Cave Adventure", January 2019, <https://en.wikipedia.org/wiki/Colossal_Cave_Adventure>.

[draft-arkko-arch-internet-threat-model]
Arkko, J., "Changes in the Internet Threat Model", n.d., <https://datatracker.ietf.org/doc/draft-arkko-arch-internet-threat-model/>.

[draft-farrell-etm]
Farrell, S., "We're gonna need a bigger threat model", n.d., <https://datatracker.ietf.org/doc/draft-farrell-etm/>.

[GREASE] Thomson, M., "Long-term Viability of Protocol Extension Mechanisms", July 2019, <https://tools.ietf.org/html/draft-iab-use-it-or-lose-it-00>.

[I-D.irtf-panrg-questions]
Trammell, B., "Current Open Questions in Path Aware Networking", Work in Progress, Internet-Draft, draft-irtf-panrg-questions-07, 29 August 2020, <http://www.ietf.org/internet-drafts/draft-irtf-panrg-questions-07.txt>.

[IEN-119] Forgie, J., "ST - A Proposed Internet Stream Protocol", September 1979, <https://www.rfc-editor.org/ien/ien119.txt>.

[ITAT] "IAB Workshop on Internet Technology Adoption and Transition (ITAT)", December 2013, <https://www.iab.org/activities/workshops/itat/>.

[model-t] "Model-t -- Discussions of changes in Internet deployment patterns and their impact on the Internet threat model", n.d., <https://www.iab.org/mailman/listinfo/model-t>.

[MP-TCP] "Multipath TCP Working Group Home Page", n.d., <https://datatracker.ietf.org/wg/mptcp/about/>.

[NANOG-35] "North American Network Operators Group NANOG-35 Agenda", October 2005, <https://www.nanog.org/meetings/nanog35/agenda>.

[NSIS-CHARTER-2001] "Next Steps In Signaling Working Group Charter", March 2011, <https://datatracker.ietf.org/doc/charter-ietf-nsis/>.

[PANRG] "Path Aware Networking Research Group (Home Page)", n.d., <https://irtf.org/panrg>.

[PANRG-103-Min] "Path Aware Networking Research Group - IETF-103 Minutes", November 2018, <https://datatracker.ietf.org/doc/minutes-103-panrg/>.

[PANRG-105-Min] "Path Aware Networking Research Group - IETF-105 Minutes", July 2019, <https://datatracker.ietf.org/doc/minutes-105-panrg/>.

[PANRG-106-Min] "Path Aware Networking Research Group - IETF-106 Minutes", November 2019, <https://datatracker.ietf.org/doc/minutes-106-panrg/>.

[PANRG-99] "Path Aware Networking Research Group - IETF-99", July 2017, <https://datatracker.ietf.org/meeting/99/sessions/panrg>.

[PATH-Decade] Bonaventure, O., "A Decade of Path Awareness", July 2017, <https://datatracker.ietf.org/doc/slides-99-panrg-a-decade-of-path-awareness/>.

[PathProp] Enghardt, T. and C. Kraehenbuehl, "A Vocabulary of Path Properties", November 2019, <https://tools.ietf.org/html/draft-enghardt-panrg-path-properties-03>.

[QS-SAT] Secchi, R., Sathiaseelan, A., Potorti, F., Gotta, A., and G. Fairhurst, "Using Quick-Start to enhance TCP-friendly rate control performance in bidirectional satellite networks", 2009, <https://dl.acm.org/citation.cfm?id=3160304.3160305>.

[QUIC-WG] "QUIC Working Group Home Page", n.d., <https://datatracker.ietf.org/wg/quic/about/>.

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, DOI 10.17487/RFC0792, September 1981, <https://www.rfc-editor.org/info/rfc792>.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <https://www.rfc-editor.org/info/rfc793>.

[RFC1016] Prue, W. and J. Postel, "Something a Host Could Do with Source Quench: The Source Quench Introduced Delay (SQuID)", RFC 1016, DOI 10.17487/RFC1016, July 1987, <https://www.rfc-editor.org/info/rfc1016>.

[RFC1190] Topolcic, C., "Experimental Internet Stream Protocol: Version 2 (ST-II)", RFC 1190, DOI 10.17487/RFC1190, October 1990, <https://www.rfc-editor.org/info/rfc1190>.

[RFC1633] Braden, R., Clark, D., and S. Shenker, "Integrated Services in the Internet Architecture: an Overview", RFC 1633, DOI 10.17487/RFC1633, June 1994, <https://www.rfc-editor.org/info/rfc1633>.

[RFC1809] Partridge, C., "Using the Flow Label Field in IPv6", RFC 1809, DOI 10.17487/RFC1809, June 1995, <https://www.rfc-editor.org/info/rfc1809>.

[RFC1819] Delgrossi, L., Ed. and L. Berger, Ed., "Internet Stream Protocol Version 2 (ST2) Protocol Specification - Version ST2+", RFC 1819, DOI 10.17487/RFC1819, August 1995, <https://www.rfc-editor.org/info/rfc1819>.

[RFC1887] Rekhter, Y., Ed. and T. Li, Ed., "An Architecture for IPv6 Unicast Address Allocation", RFC 1887, DOI 10.17487/RFC1887, December 1995, <https://www.rfc-editor.org/info/rfc1887>.

[RFC2001] Stevens, W., "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms", RFC 2001, DOI 10.17487/RFC2001, January 1997, <https://www.rfc-editor.org/info/rfc2001>.

[RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <https://www.rfc-editor.org/info/rfc2205>.

[RFC2210] Wroclawski, J., "The Use of RSVP with IETF Integrated Services", RFC 2210, DOI 10.17487/RFC2210, September 1997, <https://www.rfc-editor.org/info/rfc2210>.

[RFC2211] Wroclawski, J., "Specification of the Controlled-Load Network Element Service", RFC 2211, DOI 10.17487/RFC2211, September 1997, <https://www.rfc-editor.org/info/rfc2211>.

[RFC2212] Shenker, S., Partridge, C., and R. Guerin, "Specification of Guaranteed Quality of Service", RFC 2212, DOI 10.17487/RFC2212, September 1997, <https://www.rfc-editor.org/info/rfc2212>.

[RFC2215] Shenker, S. and J. Wroclawski, "General Characterization Parameters for Integrated Service Network Elements", RFC 2215, DOI 10.17487/RFC2215, September 1997, <https://www.rfc-editor.org/info/rfc2215>.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <https://www.rfc-editor.org/info/rfc2475>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC3697] Rajahalme, J., Conta, A., Carpenter, B., and S. Deering, "IPv6 Flow Label Specification", RFC 3697, DOI 10.17487/RFC3697, March 2004, <https://www.rfc-editor.org/info/rfc3697>.

[RFC4094] Manner, J. and X. Fu, "Analysis of Existing Quality-of-Service Signaling Protocols", RFC 4094, DOI 10.17487/RFC4094, May 2005, <https://www.rfc-editor.org/info/rfc4094>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RFC4782] Floyd, S., Allman, M., Jain, A., and P. Sarolahti, "Quick-Start for TCP and IP", RFC 4782, DOI 10.17487/RFC4782, January 2007, <https://www.rfc-editor.org/info/rfc4782>.

[RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., Ed., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, DOI 10.17487/RFC5082, October 2007, <https://www.rfc-editor.org/info/rfc5082>.

[RFC5218] Thaler, D. and B. Aboba, "What Makes for a Successful Protocol?", RFC 5218, DOI 10.17487/RFC5218, July 2008, <https://www.rfc-editor.org/info/rfc5218>.

[RFC5533] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming Shim Protocol for IPv6", RFC 5533, DOI 10.17487/RFC5533, June 2009, <https://www.rfc-editor.org/info/rfc5533>.

[RFC5575] Marques, P., Sheth, N., Raszuk, R., Greene, B., Mauch, J., and D. McPherson, "Dissemination of Flow Specification Rules", RFC 5575, DOI 10.17487/RFC5575, August 2009, <https://www.rfc-editor.org/info/rfc5575>.

[RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, DOI 10.17487/RFC5681, September 2009, <https://www.rfc-editor.org/info/rfc5681>.

[RFC5971] Schulzrinne, H. and R. Hancock, "GIST: General Internet Signalling Transport", RFC 5971, DOI 10.17487/RFC5971, October 2010, <https://www.rfc-editor.org/info/rfc5971>.

[RFC5973] Stiemerling, M., Tschofenig, H., Aoun, C., and E. Davies, "NAT/Firewall NSIS Signaling Layer Protocol (NSLP)", RFC 5973, DOI 10.17487/RFC5973, October 2010, <https://www.rfc-editor.org/info/rfc5973>.

[RFC5974] Manner, J., Karagiannis, G., and A. McDonald, "NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling", RFC 5974, DOI 10.17487/RFC5974, October 2010, <https://www.rfc-editor.org/info/rfc5974>.

[RFC5981] Manner, J., Stiemerling, M., Tschofenig, H., and R. Bless, Ed., "Authorization for NSIS Signaling Layer Protocols", RFC 5981, DOI 10.17487/RFC5981, February 2011, <https://www.rfc-editor.org/info/rfc5981>.

[RFC6294] Hu, Q. and B. Carpenter, "Survey of Proposed Use Cases for the IPv6 Flow Label", RFC 6294, DOI 10.17487/RFC6294, June 2011, <https://www.rfc-editor.org/info/rfc6294>.

[RFC6398] Le Faucheur, F., Ed., "IP Router Alert Considerations and Usage", BCP 168, RFC 6398, DOI 10.17487/RFC6398, October 2011, <https://www.rfc-editor.org/info/rfc6398>.

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, DOI 10.17487/RFC6437, November 2011, <https://www.rfc-editor.org/info/rfc6437>.

[RFC6438] Carpenter, B. and S. Amante, "Using the IPv6 Flow Label for Equal Cost Multipath Routing and Link Aggregation in Tunnels", RFC 6438, DOI 10.17487/RFC6438, November 2011, <https://www.rfc-editor.org/info/rfc6438>.

[RFC6633] Gont, F., "Deprecation of ICMP Source Quench Messages", RFC 6633, DOI 10.17487/RFC6633, May 2012, <https://www.rfc-editor.org/info/rfc6633>.

[RFC6928] Chu, J., Dukkipati, N., Cheng, Y., and M. Mathis, "Increasing TCP's Initial Window", RFC 6928, DOI 10.17487/RFC6928, April 2013, <https://www.rfc-editor.org/info/rfc6928>.

[RFC7305] Lear, E., Ed., "Report from the IAB Workshop on Internet Technology Adoption and Transition (ITAT)", RFC 7305, DOI 10.17487/RFC7305, July 2014, <https://www.rfc-editor.org/info/rfc7305>.

[RFC7418] Dawkins, S., Ed., "An IRTF Primer for IETF Participants", RFC 7418, DOI 10.17487/RFC7418, December 2014, <https://www.rfc-editor.org/info/rfc7418>.

[RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085, March 2017, <https://www.rfc-editor.org/info/rfc8085>.

[RFC8170] Thaler, D., Ed., "Planning for Protocol Adoption and Subsequent Transitions", RFC 8170, DOI 10.17487/RFC8170, May 2017, <https://www.rfc-editor.org/info/rfc8170>.

[SAAG-105-Min]
"Security Area Open Meeting - IETF-105 Minutes", July 2019, <https://datatracker.ietf.org/meeting/105/materials/minutes-105-saag-00>.

[SAF07] Sarolahti, P., Allman, M., and S. Floyd, "Determining an appropriate sending rate over an underutilized network path", Computer Networking Volume 51, Number 7, May 2007.

[Sch11] Scharf, M., "Fast Startup Internet Congestion Control for Broadband Interactive Applications", Ph.D. Thesis, University of Stuttgart, April 2011.

[Shim6-35] Meyer, D., Huston, G., Schiller, J., and V. Gill, "IAB IPv6 Multihoming Panel at NANOG 35", NANOG North American Network Operator Group, October 2005, <https://www.youtube.com/watch?v=ji6Y_rYHAQs>.

[TRIGTRAN-55]
"Triggers for Transport BOF at IETF 55", July 2003, <https://www.ietf.org/proceedings/55/239.htm>.

[TRIGTRAN-56]
"Triggers for Transport BOF at IETF 56", November 2003, <https://www.ietf.org/proceedings/56/251.htm>.

Author's Address

Spencer Dawkins (editor)
Tencent America

Email: spencerdawkins.ietf@gmail.com

TSVWG Y. Li
INTERNET-DRAFT X. Zhou
Intended Status: Informational Huawei
Expires: March 30, 2019 September 26, 2018

Overlayed Path Segment Forwarding (OPSF) Problem Statement
draft-li-overlayed-path-segment-forwarding-ps-00

Abstract

Various overlays are used in networks including WAN, enterprise campus and others. End to end path are divided into multiple segments some of which are overlay encapsulated to achieve better path selection, lower latency and so on. Traditional end-to-end transport layer is not very responding to microburst and non-congestive packet loss caused by the different characteristics of the path segments. With the potential transport enhancement for the existing or purposely created overlayed path segment, end to end throughput can be improved. This document illustrates the problems in some use cases and tries to inspire more about whether and how to solve them by introducing a reliable, efficient and non-intrusive transport forwarding over the overlayed path segment(s).

Table of Contents

1. Introduction

Overlay tunnels are widely deployed for various networks, including long haul WAN interconnection, enterprise wireless access networks, etc. End to end connection are normally broken into multiple path segments for different purposes, for instance, selecting a better overlay path over the WAN or deliver the packets over the heterogenous networks like enterprise access and core networks.

TCP-like transport layer provides end to end flow control and congestion control for path reliability and high throughput. Such an approach has the problems of slow congestion responding and non-congestive loss misinterpretation at the sender and does not achieve the optimal performance in certain cases.

Some of the problems have been well known over years. With new technologies are emerging like NFV (Network Function Virtualization) and various flexible overlay protocols, forwarding over the specific overlayed path segment(s) can be considered to be enhanced by providing a reliable and non-intrusive transport to improve the throughput to solve those problems.

This document illustrates the problems in some use cases and tries to inspire more about whether and how to solve them by introducing a reliable, efficient and non-intrusive transport forwarding for the overlayed path segment(s).

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

OPSF: Overlayed Path Segment Forwarding

3. Use Cases and Problems

The following subsections presents use cases from different scenarios using overlay tunnels with a common need of higher performance and reliable overlayed path segment in best effort networks.

3.1 Microburst in Long Haul Network

Internet is a huge sized network of networks. The interconnections of end devices using this global network are normally provided by ISPs (Internet Service Provider). This ISP provided huge network is considered as traditional Internet. CSPs (Cloud Service Provider) are connecting their data centers using Internet or self-constructed networks/links. This expands Internet’s infrastructure and together with the original ISP’s infrastructure, forms the Internet underlay.

NFV further makes it easier to dynamically provision a new virtual node as a work load in a cloud for CPU/storage intensive functions. With the aid of various mechanisms such as kernel bypassing and Virtual IO, forwarding based on virtual node is becoming more and more effective. The interconnections among the purposely positioned virtual nodes and/or the existing nodes with vitalization functions potentially form "the Second Plane" or overlay of Internet. It is called the Cloud-Internet Overlay Network (CION) in this document.

CION makes use of overlay technologies to direct the traffic going to the specific path regardless the underlying physical topology to achieve better service delivery. Figure 1 shows an emerging multi-segment overlay over large geographic distances. It purposely creates or selects overlay nodes (ON) from clouds/Internet. Segment here is the virtual hop between two ONs. By directing the traffic to be forwarded along those virtual nodes rather than the default path, better delivery in terms of throughput and delay can be achieved. When a large number of potential virtual nodes are available, there is a high chance that a better path could be found [CRONets].

```
                  ______________
                 /  domain 1   \
                /               \
            ___/                 -------------\
           /                                   \
    PoP1 ->--ON1                                \
          |   |                            ON4------>-- PoP2
          |   |   ON2                     ___|__/
           \__|_ |->|         ______     /   |
              | \|__|__      /     \    /    |
              |  |  |  \____/       \__/     |
             \|/ |  |        ______          |
              |  |  |    ___/     \          |
              |  | \|/  /          \______   |
              |  |  |  /         domain 2 \ /|\
              |  |  | |       ON3         |  |
              |  |  |  \      |->|        |  |
              |  |  |   \_____|__|_______/   |
              | /|\ |         | \|/          |
              |  |  |         |  |           |
              |  |  |        /|\ |           |
       +--------------------------------------------------+
       |      |  |  |         |  |           |   Internet |
       |      o--o  o---o->---o  o---o->--o--o   underlay |
       +--------------------------------------------------+
```

Figure 1. Cloud-Internet Overlay Network (CION)

Microburst is an unexpected data bursts within a very small time window (probably in micro-seconds). Some research shows microbursts happen even for underutilized link [BurstyAna]. The short spikes caused by microburst result in higher jitter and sometimes packet loss in a network. Such loss may trigger the congestion control like reducing the sending rate at the TCP sender as it exhibits the normal pattern of congestion loss in terms of duplicate acknowledgements and/or RTT increases. As microburst is extremely short, the packet loss caused by it is non-persistent and rather random. Therefore it does not necessarily require the sender to reduce its sending rate. Invoking the congestion control at the sender may unnecessarily make the average sending rate low and degrades the throughput in long haul CION. In addition, long haul transmission may take hundreds of milliseconds. The packet loss response at the sender to microburst over the long haul transmission is not timely. Sender’s reaction does not really respond to the current instantaneous path situation.

Overlay nodes in the middle can potentially offer new possibilities, e.g. retransmission over ONs, to better response to microbursts. Such enhancement can be enabled based on the individual overlayed path segment rather than on the entire end to end path to improve the response time and performance from the packet loss/re-order caused by microburst. Such enhancement should avoid racing with higher layer transport protocols.

3.2 Non-congestive Loss in WiFi Accessed Campus Overlay

Different path segments have different characteristics. The probabilities of packet loss over every and each segments have a large variance. The non-congestive packet loss usually occurs in some specific overlayed path segments. End to end TCP-like transport protocols do not take this factor into careful account. It assumes that packet loss for any reason is almost evenly distributed across the entire path, and adjusts the sender to accommodate the packet loss of the bottleneck segment. This results in non-optimal sending rate in some cases.

Figure 2 shows the WiFi accessed enterprise campus. AP connects to its edge switch normally using Cat5/5e twisted-pair cable which typically provides less than 10G bandwidth. The data packets are tunneled using various overlay mechanisms, like VXLAN [RFC7348], LISP [RFC6830] or CAPWAP [RFC5415]. Two edge switches use another overlay segment over campus core network to deliver the packets which provides more functions like policy enforcement and mobility enhancement. This overlay is usually over fiber which provides higher bandwidth.

```
                 _____________________________
     +-----+  /|                             |\  +-----+
     |edge1| | |                             | | |edge2|
     +-----+  \|_____________________________|/  +-----+
       _                                           _
      /_\             fiber                       /_\
      | |                                         | |
      | |                                         | |
      | | Cat5/5e                                 | | Cat5/5e
      | | cable                                   | | cable
      |_|                                         |_|
      \_/                                         \_/
     +-----+                                     +-----+
     | AP1 |                                     | AP2 |
     +-----+                                     +-----+
        |                                           |
        | WiFi access                               | WiFi access
        |                                           |
     +----+                                       +----+
     |STA1|                                       |STA2|
     +----+                                       +----+
```

Figure 2. WiFi accessed Campus Overlay Network

Cat5/5e cables, especially UTP (Unshielded twisted pair), are susceptible to distance, interference, and bundling. The environment and the way they are deployed cause drastic changes in random loss rate. The overlay tunnel running over it will have more transmission unreliability than the overlay running on the fiber. Current transport layer is not able to identify such specific problematic segment and simply leaves it for the end to end congestion control to handle it so that the sender may be kept at a lower sending rate and the throughput is not optimal.

In addition to the uplink of the AP, the non-congestive packet loss generated by the wireless access link itself accounts for the largest proportion in the end-to-end path. Wifi access is affected by fades, interference, attenuation and corruption. Non-congestive loss is common. Its link layer has mechanisms to do the packet recovery. However the number of local link layer retransmission is usually based on the empirical value or the static configuration. When the value is not properly chosen, the TCP sender can be unnecessarily exposed by the random packet loss and reduce the sending rate. It is hard to make the link layer frame recovery work in concert with the current end to end transport layer.

3.3 Higher Reliability and Low Latency for Interactive Application

Mobile gaming and VoIP like application normally can not tolerate a retransmission even over a path segment. When two divergent overlay segments are available like shown in figure 3 for path from ON1 to ON2, purposely duplicating packets over two segments provides more reliability. Two disjoint segments can usually be obtained by measuring to find segments with very low mathematical correlation in latency change. When the number of overlay nodes is large, it is easy to find such disjoint segments. Random node or memory failure may also benefit from the duplicating packets over disjoint segments.

```
                     ON-A
             +----------o------------------+
             |                             |
             |                             |
      A -----o ON1                     ON2o----- B
             |                             |
             +-----------------------o-----+
                                   ON-B
```

Figure 3. Multiple Overlayed Path Segments for Higher Reliability

4. Features to be Considered for OPSF (Overlayed Path Segment Forwarding)

The diagram shown in Figure 4 illustrates a typical scenario with an overlayed path segment. Transport layer provide the end to end flow control between two end host. When an overlayed path segment exists or is purposely created between two overlay nodes, an enhanced forwarding over that segment can potentially solve some problems of end to end transport performance issues and at the same time provides more reliability and flexibility to traffic path.

```
                                                      ON=overlay node
                                                      UN=underlay node

    +-----------+                                          +-----------+
    |Application|<-------------- end-to-end -------------> |Application|
    +-----------+                                          +-----------+
    | Transport |<-------------- end-to-end -------------> | Transport |
    |           |                                          |           |
    +-----------+                    overlayed             +-----------+
    |           |            +----+ path segment+----+     |           |
    |           |            | ON |<----------->| ON |     |           |
    |  Network  |   +----+   +----+   +----+    +----+     |  Network  |
    |           |<->| UN |<->| UN |<->| UN |<-->| UN |<--->|           |
    +-----------+   +----+   +----+   +----+    +----+     +-----------+
       End Host                                               End Host
```

Figure 4. A Simple Overlayed Path Segment Forwarding Usage Scenario

Features need more investigations include,

- Enhancement for the overlayed path segment, like retransmission, FEC(forward error correction), duplicating packet over the segments, lightweighted congestion control, etc. When the segment is a small portion of the whole end to end path, the retransmission over it has more benefit. Retransmission over the path segment has to be carefully designed to avoid the racing condition with the upper layer. The segment enabled retransmission may measure the segment RTT by itself to determine the appropriate retransmission attempts. On the other hand, the upper layers including the applications can indicate the credit as the safe band time that allows for the overlayed path segment to do the retransmission. At the same time, the persistent congestion caused packet loss should be exposed to the upper transport layer, so that the sender's congestion control can work properly. The timing of activation of the enhancement scheme, parameters such as the threshold setting of retransmission are worthy of further determination.

- Measurement based path selection for better performance, backup or load balancing. Overlay nodes have to be continuously monitored in order to find one or more appropriate overlayed paths. Such measurement can be in-band or out of band of data packets. When more than one overlayed segment with the same ingress and egress are used, it has to be determined how the traffic are split and merged.

5. Security Considerations

TBD

6. IANA Considerations

No IANA action is required.

7. References

7.1 Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

7.2 Informative References

[RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace", RFC 4122, July 2005.

[RFC5415] Calhoun, P., Ed., Montemurro, M., Ed., and D. Stanley, Ed., "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", RFC 5415, March 2009.

[BurstyAna] Chung S., Agrawal D., Kim M., Hong J., and Park K. "Analysis of bursty packet loss characteristics on underutilized links using SNMP", IEEE/IFIP E2EMON, 2004.

[CRONets] Cai, C. X., Le, F., Sun, X., Xie, G. G., Jamjoom, H., and Campbell, R. H. CRONets: Cloud-Routed Overlay Networks. In 36th International Conference on Distributed Computing Systems (ICDCS) (2016), IEEE, pp. 67--77.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for

Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, August 2014.

Authors' Addresses

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56624584
EMail: liyizhou@huawei.com

Xingwang Zhou
Huawei Technologies
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

EMail: zhouxingwang@huawei.com

TSVWG
INTERNET-DRAFT
Intended Status: Informational
Expires: April 12, 2019

Y. Li
X. Zhou
Huawei
October 9, 2018

Overlayed Path Segment Forwarding (OPSF) Problem Statement
draft-li-tsvwg-overlayed-path-segment-fwding-ps-00

Abstract

Various overlays are used in networks including WAN, enterprise campus and others. End to end path are divided into multiple segments some of which are overlay encapsulated to achieve better path selection, lower latency and so on. Traditional end-to-end transport layer is not very responding to microburst and non-congestive packet loss caused by the different characteristics of the path segments. With the potential transport enhancement for the existing or purposely created overlayed path segment, end to end throughput can be improved. This document illustrates the problems in some use cases and tries to inspire more about whether and how to solve them by introducing a reliable, efficient and non-intrusive transport forwarding over the overlayed path segment(s).

Table of Contents

1. Introduction

Overlay tunnels are widely deployed for various networks, including long haul WAN interconnection, enterprise wireless access networks, etc. End to end connection are normally broken into multiple path segments for different purposes, for instance, selecting a better overlay path over the WAN or deliver the packets over the heterogenous networks like enterprise access and core networks.

TCP-like transport layer provides end to end flow control and congestion control for path reliability and high throughput. Such an approach has the problems of slow congestion responding and non-congestive loss misinterpretation at the sender and does not achieve the optimal performance in certain cases.

Some of the problems have been well known over years. With new technologies are emerging like NFV (Network Function Virtualization) and various flexible overlay protocols, forwarding over the specific overlayed path segment(s) can be considered to be enhanced by providing a reliable and non-intrusive transport to improve the throughput to solve those problems.

This document illustrates the problems in some use cases and tries to inspire more about whether and how to solve them by introducing a reliable, efficient and non-intrusive transport forwarding for the overlayed path segment(s).

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

OPSF: Overlayed Path Segment Forwarding

3. Use Cases and Problems

The following subsections presents use cases from different scenarios using overlay tunnels with a common need of higher performance and reliable overlayed path segment in best effort networks.

3.1 Microburst in Long Haul Network

Internet is a huge sized network of networks. The interconnections of end devices using this global network are normally provided by ISPs (Internet Service Provider). This ISP provided huge network is considered as traditional Internet. CSPs (Cloud Service Provider) are connecting their data centers using Internet or self-constructed networks/links. This expands Internet's infrastructure and together with the original ISP's infrastructure, forms the Internet underlay.

NFV further makes it easier to dynamically provision a new virtual node as a work load in a cloud for CPU/storage intensive functions. With the aid of various mechanisms such as kernel bypassing and Virtual IO, forwarding based on virtual node is becoming more and more effective. The interconnections among the purposely positioned virtual nodes and/or the existing nodes with vitalization functions potentially form "the Second Plane" or overlay of Internet. It is called the Cloud-Internet Overlay Network (CION) in this document.

CION makes use of overlay technologies to direct the traffic going to the specific path regardless the underlying physical topology to achieve better service delivery. Figure 1 shows an emerging multi-segment overlay over large geographic distances. It purposely creates or selects overlay nodes (ON) from clouds/Internet. Segment here is the virtual hop between two ONs. By directing the traffic to be forwarded along those virtual nodes rather than the default path, better delivery in terms of throughput and delay can be achieved. When a large number of potential virtual nodes are available, there is a high chance that a better path could be found [CRONets].

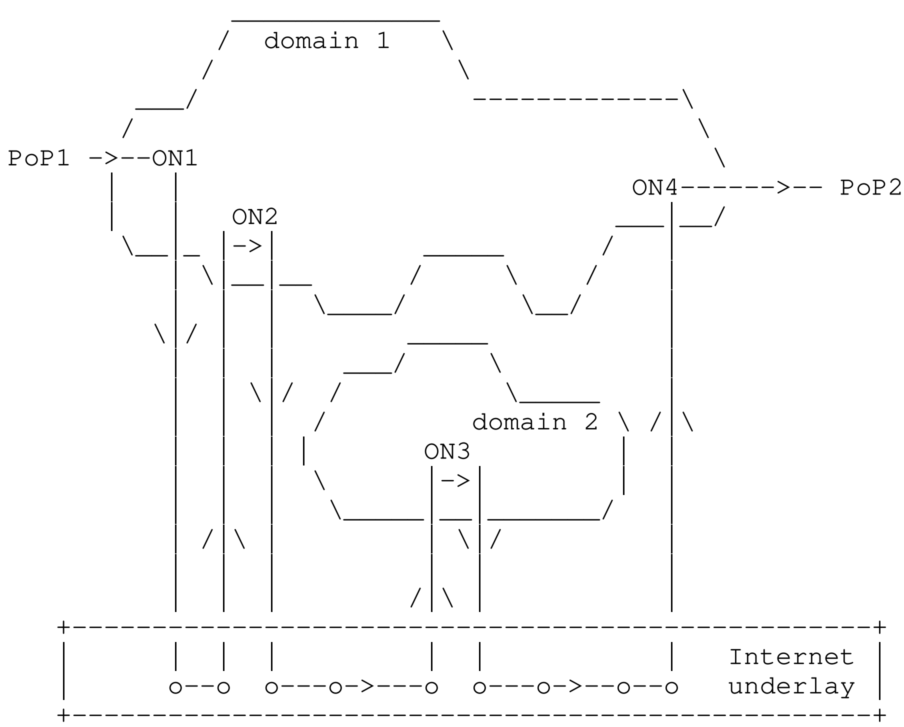

Figure 1. Cloud-Internet Overlay Network (CION)

Microburst is an unexpected data bursts within a very small time window (probably in micro-seconds). Some research shows microbursts happen even for underutilized link [BurstyAna]. The short spikes caused by microburst result in higher jitter and sometimes packet loss in a network. Such loss may trigger the congestion control like reducing the sending rate at the TCP sender as it exhibits the normal pattern of congestion loss in terms of duplicate acknowledgements and/or RTT increases. As microburst is extremely short, the packet loss caused by it is non-persistent and rather random. Therefore it does not necessarily require the sender to reduce its sending rate. Invoking the congestion control at the sender may unnecessarily make the average sending rate low and degrades the throughput in long haul CION. In addition, long haul transmission may take hundreds of milliseconds. The packet loss response at the sender to microburst over the long haul transmission is not timely. Sender's reaction does not really respond to the current instantaneous path situation.

Overlay nodes in the middle can potentially offer new possibilities, e.g. retransmission over ONs, to better response to microbursts. Such enhancement can be enabled based on the individual overlayed path segment rather than on the entire end to end path to improve the response time and performance from the packet loss/re-order caused by microburst. Such enhancement should avoid racing with higher layer transport protocols.

3.2 Non-congestive Loss in WiFi Accessed Campus Overlay

Different path segments have different characteristics. The probabilities of packet loss over every and each segments have a large variance. The non-congestive packet loss usually occurs in some specific overlayed path segments. End to end TCP-like transport protocols do not take this factor into careful account. It assumes that packet loss for any reason is almost evenly distributed across the entire path, and adjusts the sender to accommodate the packet loss of the bottleneck segment. This results in non-optimal sending rate in some cases.

Figure 2 shows the WiFi accessed enterprise campus. AP connects to its edge switch normally using Cat5/5e twisted-pair cable which typically provides less than 10G bandwidth. The data packets are tunneled using various overlay mechanisms, like VXLAN [RFC7348], LISP [RFC6830] or CAPWAP [RFC5415]. Two edge switches use another overlay segment over campus core network to deliver the packets which provides more functions like policy enforcement and mobility enhancement. This overlay is usually over fiber which provides higher bandwidth.

```
                ______________________________
   +------+  /|                              |\  +------+
   |edge1| | |                              | | |edge2|
   +------+  \|______________________________|/  +------+
     _                                              _
    /_\              fiber                         /_\
    | |                                            | |
    | |                                            | |
    | | Cat5/5e                                    | | Cat5/5e
    | | cable                                      | | cable
    |_|                                            |_|
    \_/                                            \_/
   +------+                                     +------+
   | AP1 |                                      | AP2 |
   +------+                                     +------+
      |                                            |
      | WiFi access                                | WiFi access
      |                                            |
   +-----+                                       +-----+
   |STA1|                                        |STA2|
   +-----+                                       +-----+
```

Figure 2. WiFi accessed Campus Overlay Network

Cat5/5e cables, especially UTP (Unshielded twisted pair), are susceptible to distance, interference, and bundling. The environment and the way they are deployed cause drastic changes in random loss rate. The overlay tunnel running over it will have more transmission unreliability than the overlay running on the fiber. Current transport layer is not able to identify such specific problematic segment and simply leaves it for the end to end congestion control to handle it so that the sender may be kept at a lower sending rate and the throughput is not optimal.

In addition to the uplink of the AP, the non-congestive packet loss generated by the wireless access link itself accounts for the largest proportion in the end-to-end path. Wifi access is affected by fades, interference, attenuation and corruption. Non-congestive loss is common. Its link layer has mechanisms to do the packet recovery. However the number of local link layer retransmission is usually based on the empirical value or the static configuration. When the value is not properly chosen, the TCP sender can be unnecessarily exposed by the random packet loss and reduce the sending rate. It is hard to make the link layer frame recovery work in concert with the current end to end transport layer.

3.3 Higher Reliability and Low Latency for Interactive Application

Mobile gaming and VoIP like application normally can not tolerate a retransmission even over a path segment. When two divergent overlay segments are available like shown in figure 3 for path from ON1 to ON2, purposely duplicating packets over two segments provides more reliability. Two disjoint segments can usually be obtained by measuring to find segments with very low mathematical correlation in latency change. When the number of overlay nodes is large, it is easy to find such disjoint segments. Random node or memory failure may also benefit from the duplicating packets over disjoint segments.

```
                  ON-A
             +----------o------------------+
             |                             |
             |                             |
      A -----o ON1                     ON2o----- B
             |                             |
             +-----------------------o-----+
                                    ON-B
```

Figure 3. Multiple Overlayed Path Segments for Higher Reliability

4. Features to be Considered for OPSF (Overlayed Path Segment Forwarding)

The diagram shown in Figure 4 illustrates a typical scenario with an overlayed path segment. Transport layer provide the end to end flow control between two end host. When an overlayed path segment exists or is purposely created between two overlay nodes, an enhanced forwarding over that segment can potentially solve some problems of end to end transport performance issues and at the same time provides more reliability and flexibility to traffic path.

```
                                                      ON=overlay node
                                                      UN=underlay node

    +-----------+                                          +-----------+
    |Application|<-------------- end-to-end -------------> |Application|
    +-----------+                                          +-----------+
    | Transport |<-------------- end-to-end -------------> | Transport |
    |           |                                          |           |
    +-----------+                    overlayed             +-----------+
    |           |            +----+ path segment+----+     |           |
    |           |            | ON |<----------->| ON |     |           |
    |  Network  |   +----+   +----+   +----+    +----+     |  Network  |
    |           |<->| UN |<->| UN |<->| UN |<-->| UN |<--->|           |
    +-----------+   +----+   +----+   +----+    +----+     +-----------+
       End Host                                               End Host
```

Figure 4. A Simple Overlayed Path Segment Forwarding Usage Scenario

Features need more investigations include,

- Enhancement for the overlayed path segment forwarding/transport, like retransmission, FEC(forward error correction), duplicating packet over the segments, lightweighted congestion control, etc. When the segment is a small portion of the whole end to end path, the retransmission over it has more benefit. Retransmission over the path segment has to be carefully designed to avoid the racing condition with the upper layer. The segment enabled retransmission may measure the segment RTT by itself to determine the appropriate retransmission attempts. On the other hand, the upper layers including the applications can indicate the credit as the safe band time that allows for the overlayed path segment to do the retransmission. At the same time, the persistent congestion caused packet loss should be exposed to the upper transport layer, so that the sender's congestion control can work properly. The timing of activation of the enhancement scheme, parameters such as the threshold setting of retransmission are worthy of further determination.

- Measurement based path selection for better performance, backup or load balancing. Overlay nodes have to be continuously monitored in order to find one or more appropriate overlayed paths. Such measurement can be in-band or out of band of data packets. When more than one overlayed segment with the same ingress and egress are used, it has to be determined how the traffic are split and merged.

5. Security Considerations

TBD

6. IANA Considerations

No IANA action is required.

7. References

7.1 Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

7.2 Informative References

[RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace", RFC 4122, July 2005.

[RFC5415] Calhoun, P., Ed., Montemurro, M., Ed., and D. Stanley, Ed., "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", RFC 5415, March 2009.

[BurstyAna] Chung S., Agrawal D., Kim M., Hong J., and Park K. "Analysis of bursty packet loss characteristics on underutilized links using SNMP", IEEE/IFIP E2EMON, 2004.

[CRONets] Cai, C. X., Le, F., Sun, X., Xie, G. G., Jamjoom, H., and Campbell, R. H. CRONets: Cloud-Routed Overlay Networks. In 36th International Conference on Distributed Computing Systems (ICDCS) (2016), IEEE, pp. 67--77.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for

Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, August 2014.

Authors' Addresses

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

Phone: +86-25-56624584
EMail: liyizhou@huawei.com

Xingwang Zhou
Huawei Technologies
101 Software Avenue,
Nanjing 210012
China

EMail: zhouxingwang@huawei.com