Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 22, 2023

J. Arkko
Ericsson
T. Hardie
Cisco
T. Pauly
Apple
M. Kuehlewind
Ericsson

October 19, 2022

Considerations on Application - Network Collaboration Using Path Signals

draft-iab-path-signals-collaboration-02

Abstract

This document discusses principles for designing mechanisms that use or provide path signals, and calls for standards action in specific valuable cases. RFC 8558 describes path signals as messages to or from on-path elements, and points out that visible information will be used whether it is intended as a signal or not. The principles in this document are intended as guidance for the design of explicit path signals, which are encouraged to be authenticated and include a minimal set of parties to minimize information sharing. These principles can be achieved through mechanisms like encryption of information and establishing trust relationships between entities on a path.

Status of This Memo

Table of Contents

1. Introduction

[RFC8558] defines the term "path signals" as signals to or from on-path elements. Today path signals are often implicit, e.g. derived from cleartext end-to-end information by e.g. examining transport protocols. For instance, on-path elements use various fields of the TCP header [RFC0793] to derive information about end-to-end latency as well as congestion. These techniques have evolved because the information was available and its use required no coordination with anyone. This made such techniques more easily deployable than alternative, potentially more explicit or cooperative, approaches.

However, this also means that applications and networks have often evolved their interaction without comprehensive design for how this interaction should happen or which (minimal) information would be needed for a certain function. This has led to a situation where simply information that happens to be easily available is used

instead the information that would be actually needed. As such that information may be incomplete, incorrect, or only indirectly representative of the information that is actually needed. In addition, dependencies on information and mechanisms that were designed for a different function limits the evolvability of the protocols in question.

In summary, such unplanned interactions end up having several negative effects:

- o Ossifying protocols by introducing dependencies to unintended parties that may not be updating, such as how middleboxes have limited the use of TCP options

- o Creating systemic incentives against deploying more secure or otherwise updated versions of protocols

- o Basing network behaviour on information that may be incomplete or incorrect

- o Creating a model where network entities expect to be able to use rich information about sessions passing through

For instance, features such as DNS resolution or TLS setup have been used beyond their original intent, such as in name-based filtering. MAC addresses have been used for access control, captive portals have been used to take over cleartext HTTP sessions, and so on. (This document is not about whether those practices are good or bad, it is simply stating a fact that the features were used beyond their original intent.)

Many protocol mechanisms throughout the stack fall into one of two categories: authenticated and private communication that is only visible to a very limited set of parties, often one on each "end"; and unauthenticated public communication that is also visible to all network elements on a path.

Exposed information encourages pervasive monitoring, which is described in RFC 7258 [RFC7258], and may also be used for commercial purposes, or form a basis for filtering that the applications or users do not desire. But a lack of all path signalling, on the other hand, may limit network management, debugging, or the ability for networks to optimize their services. There are many cases where elements on the network path can provide beneficial services, but only if they can coordinate with the endpoints. It also affects the ability of service providers and others to observe why problems occur [RFC9075].

As such, this situation is sometimes cast as an adversarial tradeoff between privacy and the ability for the network path to provide intended functions. However, this is perhaps an unnecessarily polarized characterization as a zero-sum situation. Not all information passing implies loss of privacy. For instance, performance information or preferences do not require disclosing the content being accessed, the user identity, or the application in use. Similarly, network congestion status information does not have to reveal network topology or the status of other users, and so on.

Increased deployment of encryption is changing this situation. Encryption provides tools for controlling information access and protects against ossification by avoiding unintended dependencies and requiring active maintenance. The increased deployment of encryption provides an opportunity to reconsider parts of Internet architecture that have used implicit derivation of input signals for on-path functions rather than explicit signalling, as recommended by RFC 8558 [RFC8558].

For instance, QUIC replaces TCP for various applications and ensures end-to-end signals are only accessible by the endpoints, ensuring evolvability [RFC9000]. QUIC does expose information dedicated for on-path elements to consume by using explicit signals for specific use cases, such as the Spin bit for latency measurements or connection ID that can be used by load balancers [I-D.ietf-quic-manageability]. This information is accessible by all on-path devices but information is limited to only those use cases. Each new use case requires additional action. This points to one way to resolve the adversity: the careful design of what information is passed.

Another extreme is to employ explicit trust and coordination between specific entities, endpoints as well as network path elements. VPNs are a good example of a case where there is an explicit authentication and negotiation with a network path element that is used to gain access to specific resources. Authentication and trust must be considered in both directions: how endpoints trust and authenticate signals from network path elements, and how network path elements trust and authenticate signals from endpoints.

The goal of improving privacy and trust on the Internet does not necessarily need to remove the ability for network elements to perform beneficial functions. We should instead improve the way that these functions are achieved and design new ways to support explicit collaboration where it is seen as beneficial. As such our goals should be:

- To ensure that information is distributed intentionally, not accidentally;

- to understand the privacy and other implications of any distributed information;

- to ensure that the information distribution is limited to the intended parties; and

- to gate the distribution of information on the participation of the relevant parties.

These goals for exposure and distribution apply equally to senders, receivers, and path elements.

Going forward, new standards work in the IETF needs to focus on addressing this gap by providing better alternatives and mechanisms for building functions that require some collaboration between endpoints and path elements.

We can establish some basic questions that any new network functions should consider:

- Which entities must consent to the information exchange?

- What is the minimum information each entity in this set needs?

- What is the effect that new signals should have?

- What is the minimum set of entities that need to be involved?

- What is the right mechanism and needed level of trust to convey this kind of information?

If we look ways network functions are achieved today, we find that many if not most of them fall short the standard set up by the questions above. Too often, they send unnecessary information or fail to limit the scope of distribution or providing any negotiation or consent.

Designing explicit signals between applications and network elements, and ensuring that all information is appropriately protected, enables information exchange in both directions that is important for improving the quality of experience and network management. The clean separation provided by explicit signals is also more conducive to protocol evolvability.

Beyond the recommendation in [RFC8558], the IAB has provided further guidance on protocol design. Among other documents, [RFC5218] provides general advice on incremental deployability based on an analysis of successes and failures and [RFC6709] discusses protocol extensibility. The Internet Technology Adoption and Transition (ITAT) workshop report [RFC7305] is also recommended reading on this same general topic. [RFC9049], an IRTF document, provides a catalogue of past issues to avoid and discusses incentives for adoption of path signals such as the need for outperforming end-to-end mechanisms or considering per-connection state.

This draft discusses different approaches for explicit collaboration and provides guidance on architectural principles to design new mechanisms. Section 2 discusses principles that good design can follow. This section also provides some examples and explanation of situations that not following the principles can lead to. Section 3 points to topics that need more to be looked at more carefully before any guidance can be given.

2. Principles

This section provides architecture-level principles for protocol designers and recommends models to apply for network collaboration and signalling.

While RFC 8558 [RFC8558] focused specifically on communication to "on-path elements", the principles described in this document apply potentially to

o on-path signalling, in either direction

o signalling with other elements in the network that are not directly on-path, but still influence end-to-end connections.

An example of on-path signalling is communication between an endpoint and a router on a network path. An example of signalling with another network element is communication between an endpoint and a network-assigned DNS server, firewall controller, or captive portal API server. Note that these communications are conceptually independent of the base flow, even if they share a packet; they are from and to other parties, rather than creating a multiparty communication.

Taken together, these principles focus on the inherent privacy and security concerns of sharing information between endpoints and network elements, emphasizing that careful scrutiny and a high bar of consent and trust need to be applied. Given the known threat of pervasive monitoring, the application of these principles is critical

to ensuring that the use of path signals does not create a disproportionate opportunity for observers to extract new data from flows.

2.1. Intentional Distribution

This guideline is best expressed in [RFC8558]:

"Fundamentally, this document recommends that implicit signals should be avoided and that an implicit signal should be replaced with an explicit signal only when the signal's originator intends that it be used by the network elements on the path. For many flows, this may result in the signal being absent but allows it to be present when needed."

The goal is that any information should be provided knowingly, for a specific purpose, sent in signals designed for that purpose, and that any use of information should be done within that purpose. And that an analysis of the security and privacy implications of the specific purpose and associated information is needed.

This guideline applies in the network element to application direction as well: a network element should not unintentionally leak information. While this document makes recommendations that are applicable to many different situations, it is important to note that the above call for careful analysis is key. Different types of information, different applications, and different directions of communication influence the the analysis, and can lead to very different conclusions about what information can be shared or with whom. For instance, it is easy to find examples of information that applications should not share with network elements (e.g., content of communications) or network elements should not share with applications (e.g., detailed user location in a wireless network). But, equally, information about other things such as the onset of congestion should be possible to share, and can be beneficial information to all parties.

Intentional distribution is a precondition for explicit collaboration enabling each entity to have the highest posssible level of control about what information to share.

2.2. Control of the Distribution of Information

Explicit signals are not enough. The entities also need to agree to exchange the information. Trust and mutual agreement between the involved entities must determine the distribution of information, in order to give adequate control to each entity over the collaboration or information sharing. This can be achieved as discussed below.

The sender needs to decide that it is willing to send information to a specific entity or set of entities. Any passing of information or request for an action needs to be explicit, and use signalling mechanisms that are designed for the purpose. Merely sending a particular kind of packet to a destination should not be interpreted as an implicit agreement.

At the same time, the recipient of information or the target of a request should have the option to agree or deny to receiving the information. It should not be burdened with extra processing if it does not have willingness or a need to do so. This happens naturally in most protocol designs, but has been a problem for some cases where "slow path" packet processing is required or implied, and the recipient or router is not willing to handle this. Performance impacts like this are best avoided, however.

In any case, all involved entities must be identified and potentially authenticated if trust is required as a prerequisite to share certain information.

Many Internet communications are not performed on behalf of the applications, but are ultimately made on behalf of users. However, not all information that may be shared directly relates to user actions or other sensitive data. All information shared must be evaluated carefully to identify potential privacy implications for users. Information that directly relates to the user should not be shared without the user's consent. It should be noted that the interests of the user and other parties, such as the application developer, may not always coincide; some applications may wish to collect more information about the user than the user would like. How to achieve a balance of control between the actual user and an application representing an user's interest is out of scope for this document.

2.3. Protecting Information and Authentication

Some simple forms of information often exist in cleartext form, e.g, ECN bits from routers are generally not authenticated or integrity protected. This is possible when the information exchanges do not carry any significantly sensitive information from the parties. Often these kind of interactions are also advisory in their nature (see also section Section 2.5).

In other cases it may be necessary to establish a secure signalling channel for communication with a specific other party, e.g., between a network element and an application. This channel may need to be authenticated, integrity protected and confidential. This is necessary, for instance, if the particular information or request

needs to be shared in confidence only with a particular, trusted network element or endpoint, or there's a danger of an attack where someone else may forge messages that could endanger the communication.

Authenticated integrity protections on signalled data can help ensure that data received in a signal has not been modified by other parties. Still, both network elements and endpoints need to be careful in processing or responding to any signal. Whether through bugs or attacks, the content of path signals can lead to unexpected behaviors or security vulnerabilities if not properly handled. As a result, the advice in Section 2.5 still applies even in situations where there's a secure channel for sending information.

However, it is important to note that authentication does not equal trust. Whether a communication is with an application server or network element that can be shown to be associated with a particular domain name, it does not follow that information about the user can be safely sent to it.

In some cases, the ability of a party to show that it is on the path can be beneficial. For instance, an ICMP error that refers to a valid flow may be more trustworthy than any ICMP error claiming to come from an address.

Other cases may require more substantial assurances. For instance, a specific trust arrangement may be established between a particular network and application. Or technologies such as confidential computing can be applied to provide an assurance that information processed by a party is handled in an appropriate manner.

2.4. Minimize Information

Each party should provide only the information that is needed for the other parties to perform the task for which collaboration is desired, and no more. This applies to information sent by an application about itself, information sent about users, or information sent by the network. This also applies to any information related to flow identification.

An architecture can follow the guideline from [RFC8558] in using explicit signals, but still fail to differentiate properly between information that should be kept private and information that should be shared. [RFC6973] also outlines this principle of data minimization as mitigation technique to protect privacy and provides further guidance.

In looking at what information can or cannot easily be passed, we need to consider both, information from the network to the application and from the application to the network.

For the application to the network direction, user-identifying information can be problematic for privacy and tracking reasons. Similarly, application identity can be problematic, if it might form the basis for prioritization or discrimination that the application provider may not wish to happen.

On the other hand, as noted above, information about general classes of applications may be desirable to be given by application providers, if it enables prioritization that would improve service, e.g., differentiation between interactive and non-interactive services.

For the network to application direction there is similarly sensitive information, such as the precise location of the user. On the other hand, various generic network conditions, predictive bandwidth and latency capabilities, and so on might be attractive information that applications can use to determine, for instance, optimal strategies for changing codecs. However, information given by the network about load conditions and so on should not form a mechanism to provide a side-channel into what other users are doing.

While information needs to be specific and provided on a per-need basis, it is often beneficial to provide declarative information that, for instance, expresses application needs than makes specific requests for action.

2.5. Limiting Impact of Information

Information shared between a network element and an endpoint of a connection needs to have a limited impact on the behavior of both endpoints and network elements. Any action that an endpoint or network element takes based on a path signal needs to be considered appropriately based on the level of authentication and trust that has been established, and be scoped to a specific network path.

For example, an ICMP signal from a network element to an endpoint can be used to influence future behavior on that particular network path (such as changing the effective packet size or closing a path-specific connection), but should not be able to cause a multipath or migration-capable transport connection to close.

In many cases, path signals can be considered to be advisory information, with the effect of optimizing or adjusting the behavior of connections on a specific path. In the case of a firewall

blocking connectivity to a given host, endpoints should only interpret that as the host being unavailable on that particular path; this is in contrast to an end-to-end authenticated signal, such as a DNSSEC-authenticated denial of existence [RFC7129].

2.6. Minimum Set of Entities

It is recommended that a design identifies the minimum number of entities needed to share a specific signal required for an identified function.

Often this will be a very limited set, such as when an application only needs to provide a signal to its peer at the other end of the connection or a host needs to contact a specific VPN gateway. In other cases a broader set is needed, such as when explicit or implicit signals from a potentially unknown set of multiple routers along the path inform the endpoints about congestion.

While it is tempting to consider removing these limitations in the context of closed, private networks, each interaction is still best considered separately, rather than simply allowing all information exchanges within the closed network. Even in a closed network with carefully managed elements there may be compromised components, as evidenced in the most extreme way by the Stuxnet worm that operated in an airgapped network. Most "closed" networks have at least some needs and means to access the rest of the Internet, and should not be modeled as if they had an impenetrable security barrier.

2.7. Carrying Information

There is a distinction between what information is sent and how it is sent. The actually sent information may be limited, while the mechanisms for sending or requesting information can be capable of sharing much more.

There is a tradeoff here between flexibility and ensuring the minimality of information in the future. The concern is that a fully generic data sharing approach between different layers and parties could potentially be misused, e.g., by making the availability of some information a requirement for passing through a network, such as making it mandatory to identify specific applications or users. This is undesirable.

This document recommends that signalling mechanisms that send information are built to specifically support sending the necessary, minimal set of information (see Section 2.4) and no more. As previously noted, flow-identifying information is a path signal in

itself, and as such provisioning of flow identifiers also requires protocol specific analysis.

Further, such mechanisms also need have an ability for establishing an agreement (see Section 2.2) and to establish sufficient trust to pass the information (see Section 2.3).

3. Further Work

This is a developing field, and it is expected that our understanding will continue to grow. One recent change is much higher use of encryption at different protocol layers. This obviously impacts the field greatly, by removing the ability to use most implicit signals. But it may also provide new tools for secure collaboration, and force a rethinking of how collaboration should be performed.

While there are some examples of modern, well-designed collaboration mechanisms, the list of examples is not long. Clearly more work is needed, if we wish to realize the potential benefits of collaboration in further cases. This requires a mindset change, a migration away from using implicit signals. And of course, we need to choose such cases where the collaboration can be performed safely, is not a privacy concern, and the incentives of the relevant parties are aligned. It should also be noted that many complex cases would require significant developments in order to become feasible.

Some of the most difficult areas are listed below. Research on these topics would be welcome. Note that the topics include both those dealing directly with on-path network element collaboration, as well as some adjacent issues that would influence such collaboration.

- Some forms of collaboration may depend on business arrangements, which may or may not be easy to put in place. For instance, some quality-of-service mechanisms involve an expectation of paying for a service. This is possible and has been successful within individual domains, e.g., users can pay for higher data rates or data caps in their ISP networks. However, it is a business-wise much harder proposition for end-to-end connections across multiple administrative domains [Claffy2015] [RFC9049].

- Secure communications with path elements is needed as discussed in Section 2.3. Finding practical ways for this has been difficult, both from the mechanics and scalability point view. And also because there is no easy way to find out which parties to trust or what trust roots would be appropriate. Some application-network element interaction designs have focused on information (such as ECN bits) that is distributed openly within a path, but there are

limited examples of designs with secure information exchange with specific network elements or endpoints.

- The use of path signals for reducing the effects of denial-of-service attacks, e.g., perhaps modern forms of "source quench" designs could be developed. The difficulty is finding a solution that would be both effective against attacks and would not enable third parties from slowing down or censoring someone else's commmunication.

- Ways of protecting information when held by network elements or servers, beyond communications security. For instance, host applications commonly share sensitive information about the user's actions with other parties, starting from basic data such as domain names learned by DNS infrastructure or source and destination addresses and protocol header information learned by all routers on the path, to detailed end user identity and other information learned by the servers. Some solutions are starting to exist for this but are not widely deployed, at least not today [Oblivious] [PDoT] [I-D.arkko-dns-confidential] [I-D.thomson-http-oblivious]. These solutions address also very specific parts of the issue, and more work remains.

- Sharing information from networks to applications. There are some working examples of this, e.g., ECN. A few other proposals have been made (see, e.g., [I-D.flinck-mobile-throughput-guidance]), but very few of those have seen deployment.

- Sharing information from applications to networks. There are a few more working examples of this (see Section 1). However, numerous proposals have been made in this space, but most of them have not progressed through standards or been deployed, for a variety of reasons [RFC9049]. Several current or recent proposals exist, however, such as [I-D.yiakoumis-network-tokens].

- Data privacy regimes generally deal with more issues than merely whether some information is shared with another party or not. For instance, there may be rules regarding how long information can be stored or what purpose information may be used for. Similar issues may also be applicable to the kind of information sharing discussed in this document.

- The present work has focused on the technical aspects of making collaboration safe and mutually beneficial, but of course, deployments need to take into account various regulatory and other policy matters. These include privacy regulation, competitive issues & network neutrality aspects, and so on.

4. Acknowledgments

The authors would like to thank everyone at the IETF, the IAB, and our day jobs for interesting thoughts and proposals in this space. Fragments of this document were also in [I-D.per-app-networking-considerations] and [I-D.arkko-path-signals-information] that were published earlier. We would also like to acknowledge [I-D.trammell-stackevo-explicit-coop] for presenting similar thoughts. Finally, the authors would like to thank Adrian Farrell, Toerless Eckert, Martin Thomson, Mark Nottingham, Luis M. Contreras, Watson Ladd, Vittorio Bertola, Andrew Campling, Eliot Lear, Spencer Dawkins, Christian Huitema, Mallory Knodel, Zhenbin Li, Chris Box, and Jeffrey Haas for useful feedback on this topic and this draft.

5. Informative References

[Claffy2015]
kc Claffy, . and D. Clark, "Adding Enhanced Services to the Internet: Lessons from History", TPRC 43: The 43rd Research Conference on Communication, Information and Internet Policy Paper , April 2015.

[I-D.arkko-dns-confidential]
Jari Arkko, and Jiri Novotny, "Privacy Improvements for DNS Resolution with Confidential Computing", draft-arkko-dns-confidential-02 (work in progress), July 2021, <https://www.ietf.org/archive/id/draft-arkko-dns-confidential-02.txt>.

[I-D.arkko-path-signals-information]
Jari Arkko, , "Considerations on Information Passed between Networks and Applications", draft-arkko-path-signals-information-00 (work in progress), February 2021, <https://www.ietf.org/archive/id/draft-arkko-path-signals-information-00.txt>.

[I-D.flinck-mobile-throughput-guidance]
Ankur Jain, , Andreas Terzis, , Hannu Flinck, , Nurit Sprecher, , Swaminathan Arunachalam, , Kevin Smith, , Vijay Devarapalli, , and Roni Bar Yanai, "Mobile Throughput Guidance Inband Signaling Protocol", draft-flinck-mobile-throughput-guidance-04 (work in progress), March 2017, <https://www.ietf.org/archive/id/draft-flinck-mobile-throughput-guidance-04.txt>.

[I-D.ietf-quic-manageability]
Mirja Kuehlewind, and Brian Trammell, "Manageability of the QUIC Transport Protocol", draft-ietf-quic-manageability-18 (work in progress), July 2022, <https://www.ietf.org/archive/id/draft-ietf-quic-manageability-18.txt>.

[I-D.per-app-networking-considerations]
Lorenzo Colitti, and Tommy Pauly, "Per-Application Networking Considerations", draft-per-app-networking-considerations-00 (work in progress), November 2020, <https://www.ietf.org/archive/id/draft-per-app-networking-considerations-00.txt>.

[I-D.thomson-http-oblivious]
Martin Thomson, and A. Christopher Wood, "Oblivious HTTP", draft-thomson-http-oblivious-02 (work in progress), August 2021, <https://www.ietf.org/archive/id/draft-thomson-http-oblivious-02.txt>.

[I-D.trammell-stackevo-explicit-coop]
Brian Trammell, , "Architectural Considerations for Transport Evolution with Explicit Path Cooperation", draft-trammell-stackevo-explicit-coop-00 (work in progress), September 2015, <https://www.ietf.org/archive/id/draft-trammell-stackevo-explicit-coop-00.txt>.

[I-D.yiakoumis-network-tokens]
Yiannis Yiakoumis, , Nick McKeown, , and Frode Sorensen, "Network Tokens", draft-yiakoumis-network-tokens-02 (work in progress), December 2020, <https://www.ietf.org/archive/id/draft-yiakoumis-network-tokens-02.txt>.

[Oblivious]
Schmitt, P., "Oblivious DNS: Practical privacy for DNS queries", Proceedings on Privacy Enhancing Technologies 2019.2: 228-244 , 2019.

[PDoT] Nakatsuka, Y., Paverd, A., and G. Tsudik, "PDoT: Private DNS-over-TLS with TEE Support", Digit. Threat.: Res. Pract., Vol. 2, No. 1, Article 3, https://dl.acm.org/doi/fullHtml/10.1145/3431171 , February 2021.

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

[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>.

[RFC6709] Carpenter, B., Aboba, B., Ed., and S. Cheshire, "Design Considerations for Protocol Extensions", RFC 6709, DOI 10.17487/RFC6709, September 2012, <https://www.rfc-editor.org/info/rfc6709>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <https://www.rfc-editor.org/info/rfc6973>.

[RFC7129] Gieben, R. and W. Mekking, "Authenticated Denial of Existence in the DNS", RFC 7129, DOI 10.17487/RFC7129, February 2014, <https://www.rfc-editor.org/info/rfc7129>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <https://www.rfc-editor.org/info/rfc7258>.

[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>.

[RFC8558] Hardie, T., Ed., "Transport Protocol Path Signals", RFC 8558, DOI 10.17487/RFC8558, April 2019, <https://www.rfc-editor.org/info/rfc8558>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

[RFC9049] Dawkins, S., Ed., "Path Aware Networking: Obstacles to Deployment (A Bestiary of Roads Not Taken)", RFC 9049, DOI 10.17487/RFC9049, June 2021, <https://www.rfc-editor.org/info/rfc9049>.

[RFC9075] Arkko, J., Farrell, S., Kuehlewind, M., and C. Perkins, "Report from the IAB COVID-19 Network Impacts Workshop 2020", RFC 9075, DOI 10.17487/RFC9075, July 2021, <https://www.rfc-editor.org/info/rfc9075>.

Authors' Addresses

Jari Arkko
Ericsson

Email: jari.arkko@ericsson.com

Ted Hardie
Cisco

Email: ted.ietf@gmail.com

Tommy Pauly
Apple

Email: tpauly@apple.com

Mirja Kuehlewind
Ericsson

Email: mirja.kuehlewind@ericsson.com