

TSVWG
Internet-Draft
Intended status: Informational
Expires: August 22, 2019

G. Fairhurst
University of Aberdeen
C. Perkins
University of Glasgow
February 18, 2019

The Impact of Transport Header Confidentiality on Network Operation and
Evolution of the Internet
[draft-ietf-tsvwg-transport-encrypt-04](#)

Abstract

This document describes implications of applying end-to-end encryption at the transport layer. It identifies in-network uses of transport layer header information. It then reviews the implications of developing end-to-end transport protocols that use authentication to protect the integrity of transport information or encryption to provide confidentiality of the transport protocol header and expected implications of transport protocol design and network operation. Since transport measurement and analysis of the impact of network characteristics have been important to the design of current transport protocols, it also considers the impact on transport and application evolution.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 22, 2019.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1.](#) Introduction [2](#)
- [2.](#) Context and Rationale [3](#)
- [3.](#) Current uses of Transport Headers within the Network [10](#)
 - [3.1.](#) Observing Transport Information in the Network [10](#)
 - [3.2.](#) Transport Measurement [16](#)
 - [3.3.](#) Use for Network Diagnostics and Troubleshooting [20](#)
 - [3.4.](#) Header Compression [21](#)
- [4.](#) Encryption and Authentication of Transport Headers [21](#)
- [5.](#) Addition of Transport Information to Network-Layer Protocol Headers [25](#)
- [6.](#) Implications of Protecting the Transport Headers [26](#)
 - [6.1.](#) Independent Measurement [26](#)
 - [6.2.](#) Characterising "Unknown" Network Traffic [28](#)
 - [6.3.](#) Accountability and Internet Transport Protocols [28](#)
 - [6.4.](#) Impact on Operational Cost [29](#)
 - [6.5.](#) Impact on Research, Development and Deployment [30](#)
- [7.](#) Conclusions [30](#)
- [8.](#) Security Considerations [33](#)
- [9.](#) IANA Considerations [35](#)
- [10.](#) Acknowledgements [35](#)
- [11.](#) Informative References [35](#)
- [Appendix A.](#) Revision information [42](#)
- Authors' Addresses [43](#)

[1.](#) Introduction

There is increased interest in, and deployment of, new protocols that employ end-to-end encryption at the transport layer, including the transport layer headers. An example of such a transport is the QUIC transport protocol [[I-D.ietf-quic-transport](#)], currently being standardised in the IETF. Encryption of transport layer headers and payload data has many benefits in terms of protecting user privacy. These benefits have been widely discussed [[RFC7258](#)], [[RFC7624](#)], and this document strongly supports the increased use of encryption in transport protocols. There are also, however, some costs, in that the widespread use of transport encryption requires changes to

network operations, and complicates network measurement for research, operational, and standardisation purposes.

This document discusses some consequences of applying end-to-end encryption at the transport layer. It reviews the implications of developing end-to-end transport protocols that use encryption to provide confidentiality of the transport protocol header, and considers the effect of such changes on transport protocol design and network operations. It also considers anticipated implications on transport and application evolution.

Transports are increasingly encrypting and authenticating the payload (i.e., the application data carried within the transport connection) end-to-end. Such protection is encouraged, and the implications are not further discussed in this memo.

2. Context and Rationale

The transport layer provides end-to-end interactions between endpoints (processes) using an Internet path. Transport protocols layer directly over the network-layer service and are sent in the payload of network-layer packets. They support end-to-end communication between applications, supported by higher-layer protocols, running on the end systems (or transport endpoints). This simple architectural view hides one of the core functions of the transport, however, to discover and adapt to the properties of the Internet path that is currently being used. The design of Internet transport protocols is as much about trying to avoid the unwanted side effects of congestion on a flow and other capacity-sharing flows, avoiding congestion collapse, adapting to changes in the path characteristics, etc., as it is about end-to-end feature negotiation, flow control and optimising for performance of a specific application.

To achieve stable Internet operations the IETF transport community has to date relied heavily on measurement and insights of the network operations community to understand the trade-offs, and to inform selection of appropriate mechanisms, to ensure a safe, reliable, and robust Internet (e.g., [[RFC1273](#)]). In turn, the network operations community relies on being able to understand the pattern and requirements of traffic passing over the Internet, both in aggregate and at the flow level.

There are many motivations for deploying encrypted transports [[RFC7624](#)] (i.e., transport protocols that use encryption to provide confidentiality of some or all of the transport-layer header information), and encryption of transport payloads (i.e. Confidentiality of the payload data). The increasing public concerns

about interference with Internet traffic have led to a rapidly expanding deployment of encryption to protect end-user privacy, e.g., QUIC [[I-D.ietf-quic-transport](#)]. Encryption is also expected to form a basis of future transport protocol designs.

Some network operators and access providers have come to rely on the in-network measurement of transport properties and the functionality provided by middleboxes to both support network operations and enhance performance. There can therefore be implications when working with encrypted transport protocols that hide transport header information from the network. These present architectural challenges and considerations in the way transport protocols are designed, and ability to characterise and compare different transport solutions [[Measure](#)]. Implementations of network devices are encouraged to avoid side-effects when protocols are updated. Introducing cryptographic integrity checks to header fields can also prevent undetected manipulation of the field by network devices, or undetected addition of information to a packet. However, this does not prevent inspection of the information by a device on path, and it is possible that such devices could develop mechanisms that rely on the presence of such a field, or a known value in the field.

Reliance on the presence and semantics of specific header information leads to ossification. An endpoint could be required to supply a specific header to receive the network service that it desires. In some cases, this could be benign or advantageous to the protocol (e.g., recognising the start of a connection, or explicitly exposing protocol information can be expected to provide more consistent decisions by on-path devices than the use of diverse methods to infer semantics from other flow properties); in other cases this is not beneficial (e.g., a mechanism implemented in a network device, such as a firewall, that required a header field to have only a specific known set of values could prevent the device from forwarding packets using a different version of a protocol that introduces a new feature that changes the value present in this field, preventing evolution of the protocol). Experience developing Transport Layer Security [[RFC8446](#)], required a design that recognised that deployed middleboxes relied on the exposed information in TLS 1.2

Examples of the impact of ossification on transport protocol design and ease of deployment can be seen in the case of Multipath TCP (MPTCP) and the TCP Fast Open option. The design of MPTCP had to be revised to account for middleboxes, so called "TCP Normalizers", that monitor the evolution of the window advertised in the TCP headers and that reset connections if the window does not grow as expected. Similarly, TCP Fast Open has had issues with middleboxes that remove unknown TCP options, that drop segments with unknown TCP options, that drop segments that contain data and have the SYN bit set, that

drop packets with SYN/ACK that acknowledge data, or that disrupt connections that send data before the three-way handshake completes. In both cases, the issue was caused by middleboxes that had a hard-coded understanding of transport behaviour, and that interacted poorly with transports that tried to change that behaviour. Other examples have included middleboxes that rewrite TCP sequence and acknowledgement numbers but are unaware of the (newer) SACK option and don't correctly rewrite selective acknowledgements to match the changes made to the fixed TCP header.

A protocol design that uses header encryption can provide confidentiality of some or all of the protocol header information. Encryption with secure key distribution prevents an on-path device from observing the header field. It therefore prevents mechanisms being built that directly rely on the information or seek to infer semantics of an exposed header field. Using encryption to provide confidentiality of the transport layer brings some well-known privacy and security benefits and can therefore help reduce ossification of the transport layer. In particular, it is important that protocols either do not expose information where the usage could change in future protocols, or that methods that utilise the information are robust to potential changes as protocols evolve over time. To avoid unwanted inspection, a protocol could also intentionally vary the format and/or value of header fields (sometimes known as Greasing [[I-D.thomson-quic-grease](#)]). However, while encryption hides the protocol header information, it does not prevent ossification of the network service. People seeking understanding of network traffic could come to rely on pattern inferences and other heuristics as the basis for network decision and to derive measurement data, creating new dependencies on the transport protocol.

Specification of non-encrypted transport header fields explicitly allows protocol designers to make specific header information observable in the network. This supports other uses of this information by on-path devices, and at the same time this can be expected to lead to ossification of the transport header, because network forwarding could evolve to depend on the presence and/or value of these fields. The decision about which transport headers fields are made observable offers trade-offs around authentication and confidentiality versus observability, network operations and management, and ossification. For example, a design that provides confidentiality of protocol header information can impact the following activities that rely on measurement and analysis of traffic flows:

Network Operations and Research: Observable transport headers enable both operators and the research community to explicitly measure

and analyse protocol performance, network anomalies, and failure pathologies.

This information can help inform capacity planning, and assist in determining the need for equipment and/or configuration changes by network operators.

The data can also inform Internet engineering research, and help in the development of new protocols, methodologies, and procedures. Concealing the transport protocol header information makes the stream performance unavailable to passive observers along the path, and likely leads to the development of alternative methods to collect or infer that data (for example heuristics based on analysis of traffic patterns).

Providing confidentiality of the transport payload, but leaving some, or all, of the transport headers unencrypted, possibly with authentication, can provide many of the privacy and security benefits while supporting operations and research, but at the cost of ossifying the transport headers.

Protection from Denial of Service: Observable transport headers currently provide useful input to classify traffic and detect anomalous events (e.g., changes in application behaviour, distributed denial of service attacks). To be effective, this protection needs to be able to uniquely disambiguate unwanted traffic. An inability to separate this traffic using packet header information could result in less-efficient identification of unwanted traffic or development of different methods (e.g. rate-limiting of uncharacterised traffic).

Network Troubleshooting and Diagnostics: Encrypting transport header information eliminates the incentive for operators to troubleshoot since they cannot interpret the data. A flow experiencing packet loss or jitter looks like an unaffected flow when only observing network layer headers (if transport sequence numbers and flow identifiers are obscured). This limits understanding of the impact of packet loss or latency on the flows, or even localizing the network segment causing the packet loss or latency. Encrypted traffic could imply "don't touch" to some, and could limit a trouble-shooting response to "can't help, no trouble found". Additional mechanisms will need to be introduced to help reconstruct or replace transport-level metrics to support troubleshooting and diagnostics, but these add complexity and operational costs (e.g., in deploying additional functions in equipment or adding traffic overhead).

Network Traffic Analysis: Hiding transport protocol header information can make it harder to determine which transport protocols and features are being used across a network segment and to measure trends in the pattern of usage. This could impact the ability for an operator to anticipate the need for network upgrades and roll-out. It can also impact the on-going traffic engineering activities performed by operators (such as determining which parts of the path contribute delay, jitter or loss). While the impact could, in many cases, be small there are scenarios where operators directly support particular services (e.g., to troubleshoot issues relating to Quality of Service, QoS; the ability to perform fast re-routing of critical traffic, or support to mitigate the characteristics of specific radio links). The more complex the underlying infrastructure the more important this impact.

Open and Verifiable Network Data: Hiding transport protocol header information can reduce the range of actors that can capture useful measurement data. This limits the information sources available to the Internet community to understand the operation of new transport protocols, so preventing access to the information necessary to inform design decisions and standardisation of the new protocols and related operational practices.

The cooperating dependence of network, application, and host to provide communication performance on the Internet is uncertain when only endpoints (i.e., at user devices and within service platforms) can observe performance, and when performance cannot be independently verified by all parties. The ability of other stakeholders to review transport header traces can help develop deeper insight into performance. In the heterogeneous Internet, this helps extend the range of topologies, vendor equipment, and traffic patterns that are evaluated.

Independently captured data is important to help ensure the health of the research and development communities. It can provide input and test scenarios to support development of new transport protocol mechanisms, especially when this analysis can be based on the behaviour experienced in a diversity of deployed networks.

Independently verifiable performance metrics might also be utilised to demonstrate regulatory compliance in some jurisdictions, and to provide a basis for informing design decisions.

The last point leads us to consider the impact of hiding transport headers in the specification and development of protocols and standards. This has potential impact on:

- o Understanding Feature Interactions: An appropriate vantage point, coupled with timing information about traffic flows, provides a valuable tool for benchmarking equipment, functions, and/or configurations, and to understand complex feature interactions. An inability to observe transport protocol information can limit the ability to diagnose and explore interactions between features at different protocol layers, a side-effect of not allowing a choice of vantage point from which this information is observed.
- o Supporting Common Specifications: Transmission Control Protocol (TCP) is currently the predominant transport protocol used over Internet paths. Its many variants have broadly consistent approaches to avoiding congestion collapse, and to ensuring the stability of the Internet. Increased use of transport layer encryption can overcome ossification, allowing deployment of new transports and different types of congestion control. This flexibility can be beneficial, but it can come at the cost of fragmenting the ecosystem. There is little doubt that developers will try to produce high quality transports for their intended target uses, but it is not clear there are sufficient incentives to ensure good practice that benefits the wide diversity of requirements for the Internet community as a whole. Increased diversity, and the ability to innovate without public scrutiny, risks point solutions that optimise for specific needs, but accidentally disrupt operations of/in different parts of the network. The social contract that maintains the stability of the Internet relies on accepting common specifications.
- o Operational Practice: The network operations community relies on being able to understand the pattern and requirements of traffic passing over the Internet, both in aggregate and at the flow level. These operational practices have developed based on the information available from unencrypted transport headers. If this information is only carried in encrypted transport headers, operators will not be able to use this information directly. If operators still wish to use these practices, they may turn to more ambitious ways of discovering this information. For example, if an operator wants to know that traffic is audio traffic, and no longer has access to Session Description Protocol (SDP) session descriptions that would explicitly say a flow "is audio", the operator might use heuristics to guess that short UDP packets with regular spacing are carrying audio traffic. Operational practices aimed at guessing transport parameters are out of scope for this document, and are only mentioned here to recognize that encryption may not prevent operators from attempting to apply the same practices they used with unencrypted transport headers.

- o Compliance: Published transport specifications allow operators and regulators to check compliance. This can bring assurance to those operating networks, often avoiding the need to deploy complex techniques that routinely monitor and manage TCP/IP traffic flows (e.g., avoiding the capital and operational costs of deploying flow rate-limiting and network circuit-breaker methods [[RFC8084](#)]). When it is not possible to observe transport header information, methods are still needed to confirm that the traffic produced conforms to the expectations of the operator or developer.
- o Restricting research and development: Hiding transport information can impede independent research into new mechanisms, measurement of behaviour, and development initiatives. Experience shows that transport protocols are complicated to design and complex to deploy, and that individual mechanisms need to be evaluated while considering other mechanisms, across a broad range of network topologies and with attention to the impact on traffic sharing the capacity. If this results in reduced availability of open data, it could eliminate the independent self-checks to the standardisation process that have previously been in place from research and academic contributors (e.g., the role of the IRTF Internet Congestion Control Research Groups (ICCRG) and research publications in reviewing new transport mechanisms and assessing the impact of their experimental deployment)

In summary, there are trade-offs. On the one hand, transport protocol designers have often ignored the implications of whether the information in transport header fields can or will be used by in-network devices, and the implications this places on protocol evolution. This motivates a design that provides confidentiality of the header information. On the other hand, it can be expected that a lack of visibility of transport header information can impact the ways that protocols are deployed, standardised, and their operational support.

To achieve stable Internet operations the IETF transport community has to date relied heavily on measurement and insights of the network operations community to understand the trade-offs, and to inform selection of appropriate mechanisms, to ensure a safe, reliable, and robust Internet (e.g., [[RFC1273](#)],[[RFC2914](#)]).

The choice of whether future transport protocols encrypt their protocol headers therefore needs to be taken based not solely on security and privacy considerations, but also taking into account the impact on operations, standards, and research. As [[RFC7258](#)] notes: "Making networks unmanageable to mitigate [pervasive monitoring] is not an acceptable outcome, but ignoring [pervasive monitoring] would go against the consensus documented here. An appropriate balance

will emerge over time as real instances of this tension are considered." This balance between information exposed and information concealed ought to be carefully considered when specifying new transport protocols.

3. Current uses of Transport Headers within the Network

Despite transport headers having end-to-end meaning, some of these transport headers have come to be used in various ways within the Internet. In response to pervasive monitoring [[RFC7624](#)] revelations and the IETF consensus that "Pervasive Monitoring is an Attack" [[RFC7258](#)], efforts are underway to increase encryption of Internet traffic,. Applying confidentiality to transport header fields would affect how protocol information is used [[RFC8404](#)]. To understand these implications, it is first necessary to understand how transport layer headers are currently observed and/or modified by middleboxes within the network.

Transport protocols can be designed to encrypt or authenticate transport header fields. Authentication at the transport layer can be used to detect any changes to an immutable header field that were made by a network device along a path. The intentional modification of transport headers by middleboxes (such as Network Address Translation, NAT, or Firewalls) is not considered. Common issues concerning IP address sharing are described in [[RFC6269](#)].

3.1. Observing Transport Information in the Network

If in-network observation of transport protocol headers is needed, this requires knowledge of the format of the transport header:

- o Flows need to be identified at the level required to perform the observation;
- o The protocol and version of the header need to be visible, e.g., by defining the wire image [[I-D.trammell-wire-image](#)]. As protocols evolve over time and there could be a need to introduce new transport headers. This could require interpretation of protocol version information or connection setup information;
- o The location and syntax of any observed transport headers needs to be known. IETF transport protocols can specify this information.

The following subsections describe various ways that observable transport information has been utilised.

3.1.1. Flow Identification

Transport protocol header information (together with information in the network header), has been used to identify a flow and the connection state of the flow, together with the protocol options being used. In some usages, a low-numbered (well-known) transport port number has been used to identify a protocol (although port information alone is not sufficient to guarantee identification of a protocol, since applications can use arbitrary ports, multiple sessions can be multiplexed on a single port, and ports can be re-used by subsequent sessions).

Transport protocols, such as TCP and the Stream Control Transport Protocol (SCTP) specify a standard base header that includes sequence number information and other data, with the possibility to negotiate additional headers at connection setup, identified by an option number in the transport header. UDP-based protocols can use, but sometimes do not use, well-known port numbers. Some flows can instead be identified by observing signalling protocol data (e.g., [[RFC3261](#)], [[I-D.ietf-rtcweb-overview](#)]) or through the use of magic numbers placed in the first byte(s) of the datagram payload [[RFC7983](#)].

Flow identification is a common function. For example, performed by measurement activities, QoS classification, firewalls, Denial of Service, DOS, prevention. It becomes more complex and less easily achieved when multiplexing is used at or above the transport layer.

3.1.2. Metrics derived from Transport Layer Headers

Some actors manage their portion of the Internet by characterizing the performance of link/network segments. Passive monitoring can observe traffic that does not encrypt the transport header information to make inferences from transport headers to derive these performance metrics. A variety of open source and commercial tools have been deployed that utilise this information. The following metrics can be derived from transport header information:

Traffic Rate and Volume: Header information (e.g., sequence number and packet size) allows derivation of volume measures per-application, to characterise the traffic that uses a network segment or the pattern of network usage. This can be measured per endpoint or for an aggregate of endpoints (e.g., by an operator to assess subscriber usage). It can also be used to trigger measurement-based traffic shaping and to implement QoS support within the network and lower layers. Volume measures can be valuable for capacity planning and providing detail of trends, rather than the volume per subscriber.

Loss Rate and Loss Pattern: Flow loss rate can be derived (e.g., from transport sequence numbers) and has been used as a metric for performance assessment and to characterise transport behaviour. Understanding the location and root cause of loss can help an operator determine whether this requires corrective action. Network operators have used the variation in patterns of loss as a key performance metric, utilising this to detect changes in the offered service.

There are various causes of loss, including corruption of link frames (e.g., interference on a radio link), buffer overflow (e.g., due to congestion), policing (traffic management), buffer management (e.g., Active Queue Management, AQM [[RFC7567](#)]), and inadequate provision of traffic pre-emption. Understanding flow loss rate requires either maintaining per flow packet counters or by observing sequence numbers in transport headers. Loss can be monitored at the interface level by devices in the network. It is often valuable to understand the conditions under which packet loss occurs. This usually requires relating loss to the traffic flowing on the network node/segment at the time of loss.

Observation of transport feedback information (e.g., RTP Control Protocol (RTCP) reception reports [[RFC3550](#)], TCP SACK blocks) can increase understanding of the impact of loss and help identify cases where loss could have been wrongly identified, or the transport did not require the lost packet. It is sometimes more helpful to understand the pattern of loss, than the loss rate, because losses can often occur as bursts, rather than randomly-timed events.

Throughput and Goodput: The throughput achieved by a flow can be determined even when a flow is encrypted, providing the individual flow can be identified. Goodput [[RFC7928](#)] is a measure of useful data exchanged (the ratio of useful/total volume of traffic sent by a flow). This requires ability to differentiate loss and retransmission of packets (e.g., by observing packet sequence numbers in the TCP or the Real-time Transport Protocol, RTP, headers [[RFC3550](#)]).

Latency: Latency is a key performance metric that impacts application response time and user-perceived response time. It often indirectly impacts throughput and flow completion time. Latency determines the reaction time of the transport protocol itself, impacting flow setup, congestion control, loss recovery, and other transport mechanisms. The observed latency can have many components [[Latency](#)]. Of these, unnecessary/unwanted queuing in network buffers has often been observed as a significant factor

[[bufferbloat](#)]. Once the cause of unwanted latency has been identified, this can often be eliminated.

To measure latency across a part of a path, an observation point can measure the experienced round trip time (RTT) using packet sequence numbers, and acknowledgements, or by observing header timestamp information. Such information allows an observation point in the network to determine not only the path RTT, but also to measure the upstream and downstream contribution to the RTT. This could be used to locate a source of latency, e.g., by observing cases where the median RTT is much greater than the minimum RTT for a part of a path.

The service offered by network operators can benefit from latency information to understand the impact of deployment and tune deployed services. Latency metrics are key to evaluating and deploying AQM [[RFC7567](#)], DiffServ [[RFC2474](#)], and Explicit Congestion Notification (ECN) [[RFC3168](#)] [[RFC8087](#)]. Measurements could identify excessively large buffers, indicating where to deploy or configure AQM. An AQM method is often deployed in combination with other techniques, such as scheduling [[RFC7567](#)] [[RFC8290](#)] and although parameter-less methods are desired [[RFC7567](#)], current methods [[RFC8290](#)] [[RFC8289](#)] [[RFC8033](#)] often cannot scale across all possible deployment scenarios.

Variation in delay: Some network applications are sensitive to small changes in packet timing (jitter). Short and long-term delay variation can impact on the latency of a flow and hence the perceived quality of applications using the network (e.g., jitter metrics are often cited when characterising paths supporting real-time traffic). To assess the performance of such applications, it can be necessary to measure the variation in delay observed along a portion of the path [[RFC3393](#)] [[RFC5481](#)]. The requirements resemble those for the measurement of latency.

Flow Reordering: Significant packet reordering within a flow can impact time-critical applications and can be interpreted as loss by reliable transports. Many transport protocol techniques are impacted by reordering (e.g., triggering TCP retransmission, or re-buffering of real-time applications). Packet reordering can occur for many reasons, from equipment design to misconfiguration of forwarding rules. Since this impacts transport performance, network tools are needed to detect and measure unwanted/excessive reordering.

There have been initiatives in the IETF transport area to reduce the impact of reordering within a transport flow, possibly leading to a reduction in the requirements for preserving ordering. These

have promise to simplify network equipment design as well as the potential to improve robustness of the transport service. Measurements of reordering can help understand the present level of reordering within deployed infrastructure, and inform decisions about how to progress such mechanisms. Key performance indicators are retransmission rate, packet drop rate, sector utilisation level, a measure of reordering, peak rate, the ECN congestion experienced (CE) marking rate, etc.

Metrics have been defined that evaluate whether a network has maintained packet order on a packet-by-packet basis [[RFC4737](#)] and [[RFC5236](#)].

Techniques for measuring reordering typically observe packet sequence numbers. Some protocols provide in-built monitoring and reporting functions. Transport fields in the RTP header [[RFC3550](#)] [[RFC4585](#)] can be observed to derive traffic volume measurements and provide information on the progress and quality of a session using RTP. As with other measurement, metadata is often needed to understand the context under which the data was collected, including the time, observation point, and way in which metrics were accumulated. The RTCP protocol directly reports some of this information in a form that can be directly visible in the network. A user of summary measurement data needs to trust the source of this data and the method used to generate the summary information.

The above passively monitor transport protocol headers to derive metrics about network layer performance useful for operation and management of a network.

3.1.3. Transport use of Network Layer Header Fields

Information from the transport protocol can be used by a multi-field classifier as a part of policy framework. Policies are commonly used for management of the QoS or Quality of Experience (QoE) in resource-constrained networks and by firewalls that use the information to implement access rules (see also [section 2.2.2 of \[RFC8404\]](#)). Network-layer classification methods that rely on a multi-field classifier (e.g. Inferring QoS from the 5-tuple or choice of application protocol) are incompatible with transport protocols that encrypt the transport information. Traffic that cannot be classified, will typically receive a default treatment.

Transport information can also be explicitly set in network-layer header fields that are not encrypted. This can provide information to enable a different forwarding treatment by the network, even when a transport employs encryption to protect other header information.

On the one hand, the user of a transport that multiplexes multiple sub-flows could wish to hide the presence and characteristics of these sub-flows. On the other hand, an encrypted transport could set the network-layer information to indicate the presence of sub-flows and to reflect the network needs of individual sub-flows. There are several ways this could be done:

IP Address: Applications expose the addresses used by endpoints, and this is used in the forwarding decisions in network devices. Address and other protocol information can be used by a Multi-Field (MF) classifier to determine how traffic is treated [[RFC2475](#)], and hence the quality of experience for a flow.

Using the IPv6 Network-Layer Flow Label: A number of Standards Track and Best Current Practice RFCs (e.g., [[RFC8085](#)], [[RFC6437](#)], [[RFC6438](#)]) encourage endpoints to set the IPv6 Flow label field of the network-layer header. IPv6 "source nodes SHOULD assign each unrelated transport connection and application data stream to a new flow" [[RFC6437](#)]. A multiplexing transport could choose to use multiple Flow labels to allow the network to independently forward subflows. [RFC6437](#) provides further guidance on choosing a flow label value, stating these "should be chosen such that their bits exhibit a high degree of variability", and chosen so that "third parties should be unlikely to be able to guess the next value that a source of flow labels will choose". To promote privacy, the Flow Label assignment needs to avoid introducing linkability that a network device may observe. Once set, a label can provide information that can help inform network-layer queuing and forwarding [[RFC6438](#)](e.g. for Equal Cost Multi-Path, ECMP, routing, and Link Aggregation, LAG) [[RFC6294](#)]. [[RFC6438](#)] includes describes considerations when used with IPsec.

Using the Network-Layer Differentiated Services Code Point:

Applications can expose their delivery expectations to the network by setting the Differentiated Services Code Point (DSCP) field of IPv4 and IPv6 packets [[RFC2474](#)]. For example, WebRTC applications identify different forwarding treatments for individual sub-flows (audio vs. video) based on the value of the DSCP field [[I-D.ietf-tsvwg-rtcweb-qos](#)]). This provides explicit information to inform network-layer queuing and forwarding, rather than an operator inferring traffic requirements from transport and application headers via a multi-field classifier.

Since the DSCP value can impact the quality of experience for a flow, observations of service performance need to consider this field when a network path has support for differentiated service treatment.

Using Explicit Congestion Marking: ECN [[RFC3168](#)] is a transport mechanism that utilises the ECN field in the network-layer header. Use of ECN explicitly informs the network-layer that a transport is ECN-capable, and requests ECN treatment of the flows packets. An ECN-capable transport can offer benefits when used over a path with equipment that implements an AQM method with Congestion Experienced (CE) marking of IP packets [[RFC8087](#)], since it can react to congestion without also having to recover from lost packets.

ECN exposes the presence of congestion. The reception of CE-marked packets can be used to estimate the level of incipient congestion on the upstream portion of the path from the point of observation ([Section 2.5 of \[RFC8087\]](#)). Interpreting the marking behaviour (i.e., assessing congestion and diagnosing faults) requires context from the transport layer (such as path RTT).

AQM and ECN offer a range of algorithms and configuration options. Tools therefore need to be available to network operators and researchers to understand the implication of configuration choices and transport behaviour as use of ECN increases and new methods emerge [[RFC7567](#)].

Careful use of the network layer features can therefore help address some of the reasons why the network inspects transport protocol headers.

[3.2.](#) Transport Measurement

The common language between network operators and application/content providers/users is packet transfer performance at a layer that all can view and analyse. For most packets, this has been the transport layer, until the emergence of QUIC, with the obvious exception of Virtual Private Networks (VPNs) and IPsec.

When encryption conceals more layers in each packet, people seeking understanding of the network operation rely more on pattern inferences and other heuristics reliance on pattern inferences and accuracy suffers. For example, the traffic patterns between server and browser are dependent on browser supplier and version, even when the sessions use the same server application (e.g., web e-mail access). It remains to be seen whether more complex inferences can be mastered to produce the same monitoring accuracy (see [section 2.1.1 of \[RFC8404\]](#)).

When measurement datasets are made available by servers or client endpoints, additional metadata, such as the state of the network, is often required to interpret this data. Collecting and coordinating

such metadata is more difficult when the observation point is at a different location to the bottleneck/device under evaluation.

Packet sampling techniques can be used to scale the processing involved in observing packets on high rate links. This exports only the packet header information of (randomly) selected packets. The utility of these measurements depends on the type of bearer and number of mechanisms used by network devices. Simple routers are relatively easy to manage, a device with more complexity demands understanding of the choice of many system parameters. This level of complexity exists when several network methods are combined.

This section discusses topics concerning observation of transport flows, with a focus on transport measurement.

3.2.1. Point of Observation

On-path measurements are particularly useful for locating the source of problems, or to assess the performance of a network segment or a particular device configuration. Often issues can only be understood in the context of the other flows that share a particular path, common network device, interface port, etc. A simple example is monitoring of a network device that uses a scheduler or active queue management technique [[RFC7567](#)], where it could be desirable to understand whether the algorithms are correctly controlling latency, or if overload protection is working. This understanding implies knowledge of how traffic is assigned to any sub-queues used for flow scheduling, but can also require information about how the traffic dynamics impact active queue management, starvation prevention mechanisms, and circuit-breakers.

Sometimes multiple on-path observation points are needed. By correlating observations of headers at multiple points along the path (e.g., at the ingress and egress of a network segment), an observer can determine the contribution of a portion of the path to an observed metric, to locate a source of delay, jitter, loss, reordering, congestion marking, etc.

3.2.2. Use by Operators to Plan and Provision Networks

Traffic measurements (e.g., traffic volume, loss, latency) is used by operators to help plan deployment of new equipment and configurations in their networks. Data is also valuable to equipment vendors who want to understand traffic trends and patterns of usage as inputs to decisions about planning products and provisioning for new deployments. This measurement information can also be correlated with billing information when this is also collected by an operator.

A network operator supporting traffic that uses transport header encryption might not have access to per-flow measurement data. Trends in aggregate traffic can be observed and can be related to the endpoint addresses being used, but it may be impossible to correlate patterns in measurements with changes in transport protocols (e.g., the impact of changes in introducing a new transport protocol mechanism). This increases the dependency on other indirect sources of information to inform planning and provisioning.

3.2.3. Service Performance Measurement

Traffic measurements (e.g., traffic volume, loss, latency) can be used by various actors to help analyse the performance offered to the users of a network segment, and to inform operational practice.

While active measurements may be used within a network, passive measurements can have advantages in terms of eliminating unproductive test traffic, reducing the influence of test traffic on the overall traffic mix, and the ability to choose the point of observation (see [Section 3.2.1](#)). However, passive measurements can rely on observing transport headers which is not possible if those headers are encrypted.

3.2.4. Measuring Transport to Support Network Operations

Information provided by tools observing transport headers can help determine whether mechanisms are needed in the network to prevent flows from acquiring excessive network capacity. Operators can implement operational practices to manage traffic flows (e.g., to prevent flows from acquiring excessive network capacity under severe congestion) by deploying rate-limiters, traffic shaping or network transport circuit breakers [[RFC8084](#)].

Congestion Control Compliance of Traffic: Congestion control is a key transport function [[RFC2914](#)]. Many network operators implicitly accept that TCP traffic complies with a behaviour that is acceptable for use in the shared Internet. TCP algorithms have been continuously improved over decades, and they have reached a level of efficiency and correctness that custom application-layer mechanisms will struggle to easily duplicate [[RFC8085](#)].

A standards-compliant TCP stack provides congestion control that may therefore be judged safe for use across the Internet. Applications developed on top of well-designed transports can be expected to appropriately control their network usage, reacting when the network experiences congestion, by back-off and reduce the load placed on the network. This is the normal expected behaviour for IETF-specified transport (e.g., TCP and SCTP).

However, when anomalies are detected, tools can interpret the transport protocol header information to help understand the impact of specific transport protocols (or protocol mechanisms) on the other traffic that shares a network. An observation in the network can gain understanding of the dynamics of a flow and its congestion control behaviour. Analysing observed flows can help to build confidence that an application flow backs-off its share of the network load in the face of persistent congestion, and hence to understand whether the behaviour is appropriate for sharing limited network capacity. For example, it is common to visualise plots of TCP sequence numbers versus time for a flow to understand how a flow shares available capacity, deduce its dynamics in response to congestion, etc. The ability to identify sources that contribute excessive congestion is important to safe operation of network infrastructure, and mechanisms can inform configuration of network devices to complement the endpoint congestion avoidance mechanisms [[RFC7567](#)] [[RFC8084](#)] to avoid a portion of the network being driven into congestion collapse [[RFC2914](#)].

Congestion Control Compliance for UDP traffic: UDP provides a minimal message-passing datagram transport that has no inherent congestion control mechanisms. Because congestion control is critical to the stable operation of the Internet, applications and other protocols that choose to use UDP as a transport are required to employ mechanisms to prevent congestion collapse, avoid unacceptable contributions to jitter/latency, and to establish an acceptable share of capacity with concurrent traffic [[RFC8085](#)].

A network operator needs tools to understand if datagram flows comply with congestion control expectations and therefore whether there is a need to deploy methods such as rate-limiters, transport circuit breakers or other methods to enforce acceptable usage for the offered service.

UDP flows that expose a well-known header by specifying the format of header fields can allow information to be observed to gain understanding of the dynamics of a flow and its congestion control behaviour. For example, tools exist to monitor various aspects of the RTP and RTCP header information of real-time flows (see [Section 3.1.2](#), and the Secure RTP extensions [[RFC3711](#)] were explicitly designed to expose header information to enable such observation.

3.3. Use for Network Diagnostics and Troubleshooting

Transport header information can be useful for a variety of operational tasks [[RFC8404](#)]: to diagnose network problems, assess network provider performance, evaluate equipment/protocol performance, capacity planning, management of security threats (including denial of service), and responding to user performance questions. Sections [3.1.2](#) and [5](#) of [[RFC8404](#)] provide further examples. These tasks seldom involve the need to determine the contents of the transport payload, or other application details.

A network operator supporting traffic that uses transport header encryption can see only encrypted transport headers. This prevents deployment of performance measurement tools that rely on transport protocol information. Choosing to encrypt all the information reduces the ability of an operator to observe transport performance, and could limit the ability of network operators to trace problems, make appropriate QoS decisions, or response to other queries about the network service. For some this will be blessing, for others it may be a curse. For example, operational performance data about encrypted flows needs to be determined by traffic pattern analysis, rather than relying on traditional tools. This can impact the ability of the operator to respond to faults, it could require reliance on endpoint diagnostic tools or user involvement in diagnosing and troubleshooting unusual use cases or non-trivial problems. A key need here is for tools to provide useful information during network anomalies (e.g., significant reordering, high or intermittent loss).

Measurements can be used to monitor the health of a portion of the Internet, to provide early warning of the need to take action. They can assist in debugging and diagnosing the root causes of faults that concern a particular user's traffic. They can also be used to support post-mortem investigation after an anomaly to determine the root cause of a problem.

In some case, measurements may involve active injection of test traffic to perform a measurement. However, most operators do not have access to user equipment, therefore the point of test is normally different from the transport endpoint. Injection of test traffic can incur an additional costs in running such tests (e.g., the implications of capacity tests in a mobile network are obvious). Some active measurements (e.g., response under load or particular workloads) perturb other traffic, and could require dedicated access to the network segment. An alternative approach is to use in-network techniques that observe transport packet headers added while traffic traverses an operational networks to make the measurements. These measurements do not require the cooperation of an endpoint.

In other cases, measurement involves dissecting network traffic flows. The observed transport layer information can help identify whether the link/network tuning is effective and alert to potential problems that can be hard to derive from link or device measurements alone. The design trade-offs for radio networks are often very different to those of wired networks. A radio-based network (e.g., cellular mobile, enterprise WiFi, satellite access/back-haul, point-to-point radio) has the complexity of a subsystem that performs radio resource management, with direct impact on the available capacity, and potentially loss/reordering of packets. The impact of the pattern of loss and congestion, differs for different traffic types, correlation with propagation and interference can all have significant impact on the cost and performance of a provided service. The need for this type of information is expected to increase as operators bring together heterogeneous types of network equipment and seek to deploy opportunistic methods to access radio spectrum.

3.4. Header Compression

Header compression saves link bandwidth by compressing network and transport protocol headers on a per-hop basis. It was widely used with low bandwidth dial-up access links, and still finds application on wireless links that are subject to capacity constraints. Header compression has been specified for use with TCP/IP and RTP/UDP/IP flows [[RFC2507](#)], [[RFC2508](#)], [[RFC4995](#)].

While it is possible to compress only the network layer headers, significant bandwidth savings can be made if both the network and transport layer headers are compressed together as a single unit. The Secure RTP extensions [[RFC3711](#)] were explicitly designed to leave the transport protocol headers unencrypted, but authenticated, since support for header compression was considered important. Encrypting the transport protocol headers does not break such header compression, but does cause it to fall back to compressing only the network layer headers, with a significant reduction in efficiency. This may have operational impact.

4. Encryption and Authentication of Transport Headers

End-to-end encryption can be applied at various protocol layers. It can be applied above the transport to encrypt the transport payload. Encryption methods can hide information from an eavesdropper in the network. Encryption can also help protect the privacy of a user, by hiding data relating to user/device identity or location. Neither an integrity check nor encryption methods prevent traffic analysis, and usage needs to reflect that profiling of users, identification of location and fingerprinting of behaviour can take place even on encrypted traffic flows. Any header information that has a clear

definition in the protocol's message format(s), or is implied by that definition, and is not cryptographically confidentiality-protected can be unambiguously interpreted by on-path observers [[I-D.trammell-wire-image](#)].

There are several motivations:

- o One motive to use encryption is a response to perceptions that the network has become ossified by over-reliance on middleboxes that prevent new protocols and mechanisms from being deployed. This has led to a perception that there is too much "manipulation" of protocol headers within the network, and that designing to deploy in such networks is preventing transport evolution. In the light of this, a method that authenticates transport headers may help improve the pace of transport development, by eliminating the need to always consider deployed middleboxes [[I-D.trammell-plus-abstract-mech](#)], or potentially to only explicitly enable middlebox use for particular paths with particular middleboxes that are deliberately deployed to realise a useful function for the network and/or users[RFC3135].
- o Another motivation stems from increased concerns about privacy and surveillance. Some Internet users have valued the ability to protect identity, user location, and defend against traffic analysis, and have used methods such as IPsec Encapsulated Security Payload (ESP), Virtual Private Networks (VPNs) and other encrypted tunnel technologies. Revelations about the use of pervasive surveillance [[RFC7624](#)] have, to some extent, eroded trust in the service offered by network operators, and following the Snowden revelation in the USA in 2013 has led to an increased desire for people to employ encryption to avoid unwanted "eavesdropping" on their communications. Concerns have also been voiced about the addition of information to packets by third parties to provide analytics, customization, advertising, cross-site tracking of users, to bill the customer, or to selectively allow or block content. Whatever the reasons, there are now activities in the IETF to design new protocols that could include some form of transport header encryption (e.g., QUIC [[I-D.ietf-quic-transport](#)]).

Authentication methods (that provide integrity checks of protocols fields) have also been specified at the network layer, and this also protects transport header fields. The network layer itself carries protocol header fields that are increasingly used to help forwarding decisions reflect the need of transport protocols, such as the IPv6 Flow Label [[RFC6437](#)], DSCP, and ECN fields.

The use of transport layer authentication and encryption exposes a tussle between middlebox vendors, operators, applications developers and users.

- o On the one hand, future Internet protocols that enable large-scale encryption assist in the restoration of the end-to-end nature of the Internet by returning complex processing to the endpoints, since middleboxes cannot modify what they cannot see.
- o On the other hand, encryption of transport layer header information has implications for people who are responsible for operating networks and researchers and analysts seeking to understand the dynamics of protocols and traffic patterns.

Whatever the motives, a decision to use pervasive transport header encryption will have implications on the way in which design and evaluation is performed, and which can in turn impact the direction of evolution of the transport protocol stack. While the IETF can specify protocols, the success in actual deployment is often determined by many factors [[RFC5218](#)] that are not always clear at the time when protocols are being defined.

The following briefly reviews some security design options for transport protocols. A Survey of Transport Security Protocols [[I-D.ietf-taps-transport-security](#)] provides more details concerning commonly used encryption methods at the transport layer.

Authenticating the Transport Protocol Header: Transport layer header information can be authenticated. An integrity check that protects the immutable transport header fields, but can still expose the transport protocol header information in the clear, allowing in-network devices to observe these fields. An integrity check can not prevent in-network modification, but can prevent a receiving from accepting changes and avoid impact on the transport protocol operation.

An example transport authentication mechanism is TCP-Authentication (TCP-AO) [[RFC5925](#)]. This TCP option authenticates the IP pseudo header, TCP header, and TCP data. TCP-AO protects the transport layer, preventing attacks from disabling the TCP connection itself and provides replay protection. TCP-AO may interact with middleboxes, depending on their behaviour [[RFC3234](#)].

The IPsec Authentication Header (AH) [[RFC4302](#)] was designed to work at the network layer and authenticate the IP payload. This approach authenticates all transport headers, and verifies their integrity at the receiver, preventing in-network modification.

Secure RTP [[RFC3711](#)] is another example of a transport protocol that allows header authentication.

Greasing: Transport layer header information that is observable can be observed in the network. Protocols often provide extensibility features, reserving fields or values for use by future versions of a specification. The specification of receivers has traditionally ignored unspecified values, however in-network devices have emerged that ossify to require a certain value in a field, or re-use a field for another purpose. When the specification is later updated, it is impossible to deploy the new use of the field, and forwarding of the protocol could even become conditional on a specific header field value.

A protocol can intentionally vary the value, format, and/or presence of observable transport header fields. This behaviour, known as GREASE (Generate Random Extensions And Sustain Extensibility), is designed to avoid a network device ossifying the use of a specific observable field. Greasing seeks to ease deployment of new methods. It can be designed to prevent in-network devices utilising the information in a transport header, or can make an observation robust to a set of changing values, rather than a specific set of values.

Encrypting the Transport Payload: The transport layer payload can be encrypted to protect the content of transport segments. This leaves transport protocol header information in the clear. The integrity of immutable transport header fields could be protected by combining this with an integrity check.

Examples of encrypting the payload include Transport Layer Security (TLS) over TCP [[RFC8446](#)] [[RFC7525](#)], Datagram TLS (DTLS) over UDP [[RFC6347](#)] [[RFC7525](#)], Secure RTP [[RFC3711](#)], and TCPcrypt [[I-D.ietf-tcpinc-tcpcrypt](#)] which permits opportunistic encryption of the TCP transport payload.

Encrypting the Transport Headers and Payload: The network layer payload could be encrypted (including the entire transport header and the payload). This method provides confidentiality of the entire transport packet. It therefore does not expose any transport information to devices in the network, which also prevents modification along a network path.

One example of encryption at the network layer is use of IPsec Encapsulating Security Payload (ESP) [[RFC4303](#)] in tunnel mode. This encrypts and authenticates all transport headers, preventing visibility of the transport headers by in-network devices. Some Virtual Private Network (VPN) methods also encrypt these headers.

Selectively Encrypting Transport Headers and Payload: A transport protocol design can encrypt selected header fields, while also choosing to authenticate the entire transport header. This allows specific transport header fields to be made observable by network devices. End-to-end integrity checks can prevent an endpoint from undetected modification of the immutable transport headers.

Mutable fields in the transport header provide opportunities for middleboxes to modify the transport behaviour (e.g., the extended headers described in [[I-D.trammell-plus-abstract-mech](#)]). This considers only immutable fields in the transport headers, that is, fields that can be authenticated End-to-End across a path.

An example of a method that encrypts some, but not all, transport information is GRE-in-UDP [[RFC8086](#)] when used with GRE encryption.

Optional Encryption of Header Information: There are implications to the use of optional header encryption in the design of a transport protocol, where support of optional mechanisms can increase the complexity of the protocol and its implementation and in the management decisions that are required to use variable format fields. Instead, fields of a specific type ought to always be sent with the same level of confidentiality or integrity protection.

As seen, different transports use encryption to protect their header information to varying degrees. There is, however, a trend towards increased protection with newer transport protocols.

5. Addition of Transport Information to Network-Layer Protocol Headers

Some measurements can be made by adding additional protocol headers carrying operations, administration and management (OAM) information to packets at the ingress to a maintenance domain (e.g., an Ethernet protocol header with timestamps and sequence number information using a method such as 802.11ag or in-situ OAM [[I-D.ietf-ippm-ioam-data](#)]) and removing the additional header at the egress of the maintenance domain. This approach enables some types of measurements, but does not cover the entire range of measurements described in this document. In some cases, it can be difficult to position measurement tools at the required segments/nodes and there can be challenges in correlating the downstream/upstream information when in-band OAM data is inserted by an on-path device. This has the advantage that a single header can support all transport protocols, but there could also be less desirable implications of separating the operation of the transport protocol from the measurement framework.

Another example of a network-layer approach is the IPv6 Performance and Diagnostic Metrics (PDM) Destination Option [[RFC8250](#)]. This allows a sender to optionally include a destination option that carries header fields that can be used to observe timestamps and packet sequence numbers. This information could be authenticated by receiving transport endpoints when the information is added at the sender and visible at the receiving endpoint, although methods to do this have not currently been proposed. This method needs to be explicitly enabled at the sender.

Current measurements suggest it can be undesirable to rely on methods requiring the presence of network options or extension headers. IPv4 network options are often not supported (or are carried on a slower processing path) and some IPv6 networks are also known to drop packets that set an IPv6 header extension (e.g., [[RFC7872](#)]). Another disadvantage is that protocols that separately expose header information do not necessarily have an advantage to expose the information that is utilised by the protocol itself, and could manipulate this header information to gain an advantage from the network.

6. Implications of Protecting the Transport Headers

The choice of which fields to expose and which to encrypt is a design choice for the transport protocol. Any selective encryption method requires trading two conflicting goals for a transport protocol designer to decide which header fields to encrypt. Security work typically employs a design technique that seeks to expose only what is needed. This approach provides incentives to not reveal any information that is not necessary for the end-to-end communication. However, there can be performance and operational benefits in exposing selected information to network tools.

This section explores key implications of working with encrypted transport protocols.

[6.1. Independent Measurement](#)

Independent observation by multiple actors is important for scientific analysis. Encrypting transport header encryption changes the ability for other actors to collect and independently analyse data. Internet transport protocols employ a set of mechanisms. Some of these need to work in cooperation with the network layer - loss detection and recovery, congestion detection and congestion control, some of these need to work only end-to-end (e.g., parameter negotiation, flow-control).

The majority of present Internet applications use two well-known transport protocols, TCP and UDP. Although TCP represents the majority of current traffic, some real-time applications use UDP, and much of this traffic utilises RTP format headers in the payload of the UDP datagram. Since these protocol headers have been fixed for decades, a range of tools and analysis methods have become common and well-understood.

Protocols that expose the state information used by the transport protocol in their header information (e.g., timestamps used to calculate the RTT, packet numbers used to assess congestion and requests for retransmission) provide an incentive for the sending endpoint to provide correct information, increasing confidence that the observer understands the transport interaction with the network. For example, when TCP is used over an unencrypted network path (i.e., one that does not use IPsec or other encryption below the transport), it implicitly exposes header information that can be used for measurement at any point along the path. This information is necessary for the protocol's correct operation, therefore there is no incentive for a TCP implementation to put incorrect information in this transport header. A network device can have confidence that the well-known (and ossified) transport information represents the actual state of the endpoints.

When encryption is used to conceal some or all of the transport headers, the transport protocol chooses what information to reveal to the network about its internal state, what information to leave encrypted, and what fields to grease to protect against future ossification. Such a transport could be designed, for example, to provide summary data regarding its performance, congestion control state, etc., or to make an explicit measurement signal available. For example, a QUIC endpoint could set the spin bit to reflect to explicitly reveal a session's RTT [[I-D.ietf-quic-spin-exp](#)]).

When providing or using such information, it becomes important to consider the privacy of the user and their incentive for providing accurate and detailed information. Protocols that selectively reveal some transport state or measurement signals are choosing to establish a trust relationship with the network operators. There is no protocol mechanism that can guarantee that the information provided represents the actual transport state of the endpoints, since those endpoints can always send additional information in the encrypted part of the header, to update or replace whatever they reveal. This reduces the ability to independently measure and verify that a protocol is behaving as expected. Some operational uses need the information to contain sufficient detail to understand, and possibly reconstruct, the network traffic pattern for further testing; such

operators must gain the trust of transport protocol implementers if they are to correctly reveal such information.

For some usage a standardised endpoint-based logging format (e.g., based on Quic-Trace [[Quic-Trace](#)]) could offer an alternative to in-network measurement. Such information will have a diversity of uses - examples include developers wishing to debug/understand the transport/application protocols with which they work, to researchers seeking to spot trends, anomalies and to characterise variants of protocols. This use will need to establish the validity and provenance of the logging information (e.g., to establish how and when traces were captured).

However, endpoint logs do not provide equivalent information to in-network measurements. In particular, endpoint logs contain only a part of the information needed to understand the operation of network devices and identify issues such as link performance or capacity sharing between multiple flows. Additional information is needed to determine which equipment/links are used and the configuration of equipment along the network paths being measured.

[6.2.](#) Characterising "Unknown" Network Traffic

The patterns and types of traffic that share Internet capacity change over time as networked applications, usage patterns and protocols continue to evolve.

If "unknown" or "uncharacterised" traffic patterns form a small part of the traffic aggregate passing through a network device or segment of the network the path, the dynamics of the uncharacterised traffic may not have a significant collateral impact on the performance of other traffic that shares this network segment. Once the proportion of this traffic increases, the need to monitor the traffic and determine if appropriate safety measures need to be put in place.

Tracking the impact of new mechanisms and protocols requires traffic volume to be measured and new transport behaviours to be identified. This is especially true of protocols operating over a UDP substrate. The level and style of encryption needs to be considered in determining how this activity is performed. On a shorter timescale, information may also need to be collected to manage denial of service attacks against the infrastructure.

[6.3.](#) Accountability and Internet Transport Protocols

Information provided by tools observing transport headers can be used to classify traffic, and to limit the network capacity used by certain flows, as discussed in [Section 3.2.4](#)). Equally, operators

could use analysis of transport headers and transport flow state to demonstrate that they are not providing differential treatment to certain flows. Obfuscating or hiding this information using encryption may lead operators and maintainers of middleboxes (firewalls, etc.) to seek other methods to classify, and potentially other mechanisms to condition, network traffic.

A lack of data that reduces the level of precision with which flows can be classified also reduces the design space for conditioning mechanisms (e.g., rate limiting, circuit breaker techniques [[RFC8084](#)], or blocking of uncharacterised traffic), and this needs to be considered when evaluating the impact of designs for transport encryption [[RFC5218](#)].

6.4. Impact on Operational Cost

Many network operators currently utilise observed transport information as a part of their operational practice, and have developed tools and operational practices based around currently deployed transports and their applications. Encryption of the transport information prevents tools from directly observing this information. A variety of open source and commercial tools have been deployed that utilise this information for a variety of short and long term measurements.

The network will not break just because transport headers are encrypted, although alternative diagnostic and troubleshooting tools would need to be developed and deployed. Introducing a new protocol or application can require these tool chains and practice to be updated, and may in turn impact operational mechanisms, and policies. Each change can introduce associated costs, including the cost of collecting data, and the tooling needed to handle multiple formats (possibly as these co-exist in the network, when measurements need to span time periods during which changes are deployed, or to compare with historical data). These costs are incurred by an operator to manage the service and debug network issues.

At the time of writing, the additional operational cost of using encrypted transports is not yet well understood. Design trade-offs could mitigate these costs by explicitly choosing to expose selected information (e.g., header invariants and the spin-bit in QUIC[I-D.ietf-quic-transport]), the specification of common log formats and development of alternative approaches.

6.5. Impact on Research, Development and Deployment

Measurement has a critical role in the design of transport protocol mechanisms and their acceptance by the wider community (e.g., as a method to judge the safety for Internet deployment) and is increasingly being used to inform design decisions in networking research, during development of new mechanisms and protocols and in standardisation. Observation of pathologies are also important in understanding the interactions between cooperating protocols and network mechanism, the implications of sharing capacity with other traffic and the impact of different patterns of usage.

Evolution and the ability to understand (measure) the impact need to proceed hand-in-hand. Attention needs to be paid to the expected scale of deployment of new protocols and protocol mechanisms. Whatever the mechanism, experience has shown that it is often difficult to correctly implement combination of mechanisms [[RFC8085](#)]. These mechanisms therefore typically evolve as a protocol matures, or in response to changes in network conditions, changes in network traffic or changes to application usage.

New transport protocol formats are expected to facilitate an increased pace of transport evolution, and with it the possibility to experiment with and deploy a wide range of protocol mechanisms. There has been recent interest in a wide range of new transport methods, e.g., Larger Initial Window, Proportional Rate Reduction (PRR), congestion control methods based on measuring bottleneck bandwidth and round-trip propagation time, the introduction of AQM techniques and new forms of ECN response (e.g., Data Centre TCP, DCTP, and methods proposed for L4S). The growth and diversity of applications and protocols using the Internet also continues to expand. For each new method or application it is desirable to build a body of data reflecting its behaviour under a wide range of deployment scenarios, traffic load, and interactions with other deployed/candidate methods.

Open standards motivate a desire for this evaluation to include independent observation and evaluation of performance data, which in turn suggests control over where and when measurement samples are collected. This requires consideration of the appropriate balance between encrypting all and no transport information.

7. Conclusions

Confidentiality and strong integrity checks have properties that are being incorporated into new protocols and that have important benefits. The pace of development of transports using the WebRTC data channel and the rapid deployment of QUIC transport protocol can

both be attributed to using the combination of UDP as a substrate while providing confidentiality and authentication of the encapsulated transport headers and payload.

The traffic that can be observed by on-path network devices is a function of transport protocol design/options, network use, applications, and user characteristics. In general, when only a small proportion of the traffic has a specific (different) characteristic, such traffic seldom leads to operational concern, although the ability to measure and monitor it is less. The desire to understand the traffic and protocol interactions typically grows as the proportion of traffic increases in volume. The challenges increase when multiple instances of an evolving protocol contribute to the traffic that share network capacity.

An increased pace of evolution therefore needs to be accompanied by methods that can be successfully deployed and used across operational networks. This leads to a need for network operators (at various level (ISPs, enterprises, firewall maintainer, etc) to identify appropriate operational support functions and procedures.

Protocols that change their transport header format (wire format) or their behaviour (e.g., algorithms that are needed to classify and characterise the protocol), will require new tooling to be developed to catch-up with the changes. If the currently deployed tools and methods are no longer relevant then it may no longer be possible to correctly measure performance. This can increase the response-time after faults, and can impact the ability to manage the network resulting in traffic causing traffic to be treated inappropriately (e.g., rate limiting because of being incorrectly classified/monitored).

There are benefits in exposing consistent information to the network that avoids traffic being mis-classified and then receiving a default treatment by the network. The flow label and DSCP fields provide examples of how transport information can be made available for network-layer decisions. Extension headers could also be used to carry transport information that can inform network-layer decisions.

As a part of its design a new protocol specification therefore needs to weigh the benefits of ossifying common headers, versus the potential demerits of exposing specific information that could be observed along the network path, to provide tools to manage new variants of protocols. This can be done for the entire transport header, or by dividing header fields between those that are observable and mutable; those that are observable, but immutable; and those that are hidden/obfuscated.

Several scenarios to illustrate different ways this could evolve are provided below:

- o One scenario is when transport protocols provide consistent information to the network by intentionally exposing a part of the transport header. The design fixes the format of this information between versions of the protocol. This ossification of the transport header allows an operator to establish tooling and procedures that enable it to provide consistent traffic management as the protocol evolves. In contrast to TCP (where all protocol information is exposed), evolution of the transport is facilitated by providing cryptographic integrity checks of the transport header fields (preventing undetected middlebox changes) and encryption of other protocol information (preventing observation within the network, or providing incentives for the use of the exposed information, rather than inferring information from other characteristics of the flow traffic). The exposed transport information can be used by operators to provide troubleshooting, measurement and any necessary functions appropriate to the class of traffic (priority, retransmission, reordering, circuit breakers, etc).

- o An alternative scenario adopts different design goals, with a different outcome. A protocol that encrypts all header information forces network operators to act independently from apps/transport developments to extract the information they need to manage their network. A range of approaches could proliferate, as in current networks. Some operators can add a shim header to each packet as a flow as it crosses the network; other operators/managers could develop heuristics and pattern recognition to derive information that classifies flows and estimates quality metrics for the service being used; some could decide to rate-limit or block traffic until new tooling is in place. In many cases, the derived information can be used by operators to provide necessary functions appropriate to the class of traffic (priority, retransmission, reordering, circuit breakers, etc). Troubleshooting, and measurement becomes more difficult, and more diverse. This could require additional information beyond that visible in the packet header and when this information is used to inform decisions by on-path devices it can lead to dependency on other characteristics of the flow. In some cases, operators might need access to keying information to interpret encrypted data that they observe. Some use cases could demand use of transports that do not use encryption.

The direction in which this evolves could have significant implications on the way the Internet architecture develops. It exposes a risk that significant actors (e.g., developers and

transport designers) achieve more control of the way in which the Internet architecture develops. In particular, there is a possibility that designs could evolve to significantly benefit of customers for a specific vendor, and that communities with very different network, applications or platforms could then suffer at the expense of benefits to their vendors own customer base. In such a scenario, there could be no incentive to support other applications/products or to work in other networks leading to reduced access for new approaches.

8. Security Considerations

This document is about design and deployment considerations for transport protocols. Issues relating to security are discussed in the various sections of the document.

Authentication, confidentiality protection, and integrity protection are identified as Transport Features by [\[RFC8095\]](#). As currently deployed in the Internet, these features are generally provided by a protocol or layer on top of the transport protocol [\[I-D.ietf-taps-transport-security\]](#).

Confidentiality and strong integrity checks have properties that can also be incorporated into the design of a transport protocol. Integrity checks can protect an endpoint from undetected modification of protocol fields by network devices, whereas encryption and obfuscation or greasing can further prevent these headers being utilised by network devices. Hiding headers can therefore provide the opportunity for greater freedom to update the protocols and can ease experimentation with new techniques and their final deployment in endpoints. A protocol specification needs to weigh the benefits of ossifying common headers, versus the potential demerits of exposing specific information that could be observed along the network path to provide tools to manage new variants of protocols.

A protocol design that uses header encryption can provide confidentiality of some or all of the protocol header information. This prevents an on-path device from knowledge of the header field. It therefore prevents mechanisms being built that directly rely on the information or seeks to infer semantics of an exposed header field. Hiding headers can limit the ability to measure and characterise traffic.

Exposed transport headers are sometimes utilised as a part of the information to detect anomalies in network traffic. This can be used as the first line of defence to identify potential threats from DOS or malware and redirect suspect traffic to dedicated nodes responsible for DOS analysis, malware detection, or to perform packet

"scrubbing" (the normalization of packets so that there are no ambiguities in interpretation by the ultimate destination of the packet). These techniques are currently used by some operators to also defend from distributed DOS attacks.

Exposed transport header fields are sometimes also utilised as a part of the information used by the receiver of a transport protocol to protect the transport layer from data injection by an attacker. In evaluating this use of exposed header information, it is important to consider whether it introduces a significant DOS threat. For example, an attacker could construct a DOS attack by sending packets with a sequence number that falls within the currently accepted range of sequence numbers at the receiving endpoint, this would then introduce additional work at the receiving endpoint, even though the data in the attacking packet may not finally be delivered by the transport layer. This is sometimes known as a "shadowing attack". An attack can, for example, disrupt receiver processing, trigger loss and retransmission, or make a receiving endpoint perform unproductive decryption of packets that cannot be successfully decrypted (forcing a receiver to commit decryption resources, or to update and then restore protocol state).

One mitigation to off-path attack is to deny knowledge of what header information is accepted by a receiver or obfuscate the accepted header information, e.g., setting a non-predictable initial value for a sequence number during a protocol handshake, as in [[RFC3550](#)] and [[RFC6056](#)], or a port value that can not be predicted (see [section 5.1 of \[RFC8085\]](#)). A receiver could also require additional information to be used as a part of check before accepting packets at the transport layer (e.g., utilising a part of the sequence number space that is encrypted; or by verifying an encrypted token not visible to an attacker). This would also mitigate on-path attacks. An additional processing cost can be incurred when decryption needs to be attempted before a receiver is able to discard injected packets.

Open standards motivate a desire for this evaluation to include independent observation and evaluation of performance data, which in turn suggests control over where and when measurement samples are collected. This requires consideration of the appropriate balance between encrypting all and no transport information. Open data, and accessibility to tools that can help understand trends in application deployment, network traffic and usage patterns can all contribute to understanding security challenges.

9. IANA Considerations

XX RFC ED - PLEASE REMOVE THIS SECTION XXX

This memo includes no request to IANA.

10. Acknowledgements

The authors would like to thank Mohamed Boucadair, Spencer Dawkins, Tom Herbert, Jana Iyengar, Mirja Kuehlewind, Kyle Rose, Kathleen Moriarty, Al Morton, Chris Seal, Joe Touch, Brian Trammell, Chris Wood, and other members of the TSVWG for their comments and feedback.

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 688421. The opinions expressed and arguments employed reflect only the authors' view. The European Commission is not responsible for any use that may be made of that information.

This work has received funding from the UK Engineering and Physical Sciences Research Council under grant EP/R04144X/1.

11. Informative References

[bufferbloat]

Gettys, J. and K. Nichols, "Bufferbloat: dark buffers in the Internet. Communications of the ACM, 55(1):57-65", January 2012.

[I-D.ietf-ippm-ioam-data]

Brockners, F., Bhandari, S., Pignataro, C., Gredler, H., Leddy, J., Youell, S., Mizrahi, T., Mozes, D., Lapukhov, P., Chang, R., daniel.bernier@bell.ca, d., and J. Lemon, "Data Fields for In-situ OAM", [draft-ietf-ippm-ioam-data-03](#) (work in progress), June 2018.

[I-D.ietf-quic-spin-exp]

Trammell, B. and M. Kuehlewind, "The QUIC Latency Spin Bit", [draft-ietf-quic-spin-exp-01](#) (work in progress), October 2018.

[I-D.ietf-quic-transport]

Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport", [draft-ietf-quic-transport-14](#) (work in progress), August 2018.

[I-D.ietf-rtcweb-overview]

Alvestrand, H., "Overview: Real Time Protocols for Browser-based Applications", [draft-ietf-rtcweb-overview-19](#) (work in progress), November 2017.

[I-D.ietf-taps-transport-security]

Pauly, T., Perkins, C., Rose, K., and C. Wood, "A Survey of Transport Security Protocols", [draft-ietf-taps-transport-security-02](#) (work in progress), June 2018.

[I-D.ietf-tcpinc-tcpscrypt]

Bittau, A., Giffin, D., Handley, M., Mazieres, D., Slack, Q., and E. Smith, "Cryptographic protection of TCP Streams (tcpscrypt)", [draft-ietf-tcpinc-tcpscrypt-12](#) (work in progress), June 2018.

[I-D.ietf-tsvwg-rtcweb-qos]

Jones, P., Dhesikan, S., Jennings, C., and D. Druta, "DSCP Packet Markings for WebRTC QoS", [draft-ietf-tsvwg-rtcweb-qos-18](#) (work in progress), August 2016.

[I-D.thomson-quic-grease]

Thomson, M., "More Apparent Randomization for QUIC", [draft-thomson-quic-grease-00](#) (work in progress), December 2017.

[I-D.trammell-plus-abstract-mech]

Trammell, B., "Abstract Mechanisms for a Cooperative Path Layer under Endpoint Control", [draft-trammell-plus-abstract-mech-00](#) (work in progress), September 2016.

[I-D.trammell-wire-image]

Trammell, B. and M. Kuehlewind, "The Wire Image of a Network Protocol", [draft-trammell-wire-image-04](#) (work in progress), April 2018.

[Latency] Briscoe, B., "Reducing Internet Latency: A Survey of Techniques and Their Merits, IEEE Comm. Surveys & Tutorials. 26;18(3) p2149-2196", November 2014.

[Measure] Fairhurst, G., Kuehlewind, M., and D. Lopez, "Measurement-based Protocol Design, Eur. Conf. on Networks and Communications, Oulu, Finland.", June 2017.

[Quic-Trace]

"https:QUIC trace utilities //github.com/google/quic-trace".

- [RFC1273] Schwartz, M., "Measurement Study of Changes in Service-Level Reachability in the Global TCP/IP Internet: Goals, Experimental Design, Implementation, and Policy Considerations", [RFC 1273](#), DOI 10.17487/RFC1273, November 1991, <<https://www.rfc-editor.org/info/rfc1273>>.
- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", [RFC 2474](#), DOI 10.17487/RFC2474, December 1998, <<https://www.rfc-editor.org/info/rfc2474>>.
- [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>>.
- [RFC2507] Degermark, M., Nordgren, B., and S. Pink, "IP Header Compression", [RFC 2507](#), DOI 10.17487/RFC2507, February 1999, <<https://www.rfc-editor.org/info/rfc2507>>.
- [RFC2508] Casner, S. and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links", [RFC 2508](#), DOI 10.17487/RFC2508, February 1999, <<https://www.rfc-editor.org/info/rfc2508>>.
- [RFC2914] Floyd, S., "Congestion Control Principles", [BCP 41](#), [RFC 2914](#), DOI 10.17487/RFC2914, September 2000, <<https://www.rfc-editor.org/info/rfc2914>>.
- [RFC3135] Border, J., Kojo, M., Griner, J., Montenegro, G., and Z. Shelby, "Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations", [RFC 3135](#), DOI 10.17487/RFC3135, June 2001, <<https://www.rfc-editor.org/info/rfc3135>>.
- [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>>.
- [RFC3234] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and Issues", [RFC 3234](#), DOI 10.17487/RFC3234, February 2002, <<https://www.rfc-editor.org/info/rfc3234>>.

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", [RFC 3261](#), DOI 10.17487/RFC3261, June 2002, <<https://www.rfc-editor.org/info/rfc3261>>.
- [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>>.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, [RFC 3550](#), DOI 10.17487/RFC3550, July 2003, <<https://www.rfc-editor.org/info/rfc3550>>.
- [RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", [RFC 3711](#), DOI 10.17487/RFC3711, March 2004, <<https://www.rfc-editor.org/info/rfc3711>>.
- [RFC4302] Kent, S., "IP Authentication Header", [RFC 4302](#), DOI 10.17487/RFC4302, December 2005, <<https://www.rfc-editor.org/info/rfc4302>>.
- [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", [RFC 4303](#), DOI 10.17487/RFC4303, December 2005, <<https://www.rfc-editor.org/info/rfc4303>>.
- [RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", [RFC 4585](#), DOI 10.17487/RFC4585, July 2006, <<https://www.rfc-editor.org/info/rfc4585>>.
- [RFC4737] Morton, A., Ciavattone, L., Ramachandran, G., Shalunov, S., and J. Perser, "Packet Reordering Metrics", [RFC 4737](#), DOI 10.17487/RFC4737, November 2006, <<https://www.rfc-editor.org/info/rfc4737>>.
- [RFC4995] Jonsson, L-E., Pelletier, G., and K. Sandlund, "The RObusT Header Compression (ROHC) Framework", [RFC 4995](#), DOI 10.17487/RFC4995, July 2007, <<https://www.rfc-editor.org/info/rfc4995>>.
- [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>>.

- [RFC5236] Jayasumana, A., Piratla, N., Banka, T., Bare, A., and R. Whitner, "Improved Packet Reordering Metrics", [RFC 5236](#), DOI 10.17487/RFC5236, June 2008, <<https://www.rfc-editor.org/info/rfc5236>>.
- [RFC5481] Morton, A. and B. Claise, "Packet Delay Variation Applicability Statement", [RFC 5481](#), DOI 10.17487/RFC5481, March 2009, <<https://www.rfc-editor.org/info/rfc5481>>.
- [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", [RFC 5925](#), DOI 10.17487/RFC5925, June 2010, <<https://www.rfc-editor.org/info/rfc5925>>.
- [RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-Protocol Port Randomization", [BCP 156](#), [RFC 6056](#), DOI 10.17487/RFC6056, January 2011, <<https://www.rfc-editor.org/info/rfc6056>>.
- [RFC6269] Ford, M., Ed., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", [RFC 6269](#), DOI 10.17487/RFC6269, June 2011, <<https://www.rfc-editor.org/info/rfc6269>>.
- [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>>.
- [RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", [RFC 6347](#), DOI 10.17487/RFC6347, January 2012, <<https://www.rfc-editor.org/info/rfc6347>>.
- [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>>.
- [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>>.

- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", [BCP 195](#), [RFC 7525](#), DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC7567] Baker, F., Ed. and G. Fairhurst, Ed., "IETF Recommendations Regarding Active Queue Management", [BCP 197](#), [RFC 7567](#), DOI 10.17487/RFC7567, July 2015, <<https://www.rfc-editor.org/info/rfc7567>>.
- [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>>.
- [RFC7872] Gont, F., Linkova, J., Chown, T., and W. Liu, "Observations on the Dropping of Packets with IPv6 Extension Headers in the Real World", [RFC 7872](#), DOI 10.17487/RFC7872, June 2016, <<https://www.rfc-editor.org/info/rfc7872>>.
- [RFC7928] Kuhn, N., Ed., Natarajan, P., Ed., Khademi, N., Ed., and D. Ros, "Characterization Guidelines for Active Queue Management (AQM)", [RFC 7928](#), DOI 10.17487/RFC7928, July 2016, <<https://www.rfc-editor.org/info/rfc7928>>.
- [RFC7983] Petit-Huguenin, M. and G. Salgueiro, "Multiplexing Scheme Updates for Secure Real-time Transport Protocol (SRTP) Extension for Datagram Transport Layer Security (DTLS)", [RFC 7983](#), DOI 10.17487/RFC7983, September 2016, <<https://www.rfc-editor.org/info/rfc7983>>.
- [RFC8033] Pan, R., Natarajan, P., Baker, F., and G. White, "Proportional Integral Controller Enhanced (PIE): A Lightweight Control Scheme to Address the Bufferbloat Problem", [RFC 8033](#), DOI 10.17487/RFC8033, February 2017, <<https://www.rfc-editor.org/info/rfc8033>>.
- [RFC8084] Fairhurst, G., "Network Transport Circuit Breakers", [BCP 208](#), [RFC 8084](#), DOI 10.17487/RFC8084, March 2017, <<https://www.rfc-editor.org/info/rfc8084>>.
- [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>>.

- [RFC8086] Yong, L., Ed., Crabbe, E., Xu, X., and T. Herbert, "GRE-in-UDP Encapsulation", [RFC 8086](#), DOI 10.17487/RFC8086, March 2017, <<https://www.rfc-editor.org/info/rfc8086>>.
- [RFC8087] Fairhurst, G. and M. Welzl, "The Benefits of Using Explicit Congestion Notification (ECN)", [RFC 8087](#), DOI 10.17487/RFC8087, March 2017, <<https://www.rfc-editor.org/info/rfc8087>>.
- [RFC8095] Fairhurst, G., Ed., Trammell, B., Ed., and M. Kuehlewind, Ed., "Services Provided by IETF Transport Protocols and Congestion Control Mechanisms", [RFC 8095](#), DOI 10.17487/RFC8095, March 2017, <<https://www.rfc-editor.org/info/rfc8095>>.
- [RFC8250] Elkins, N., Hamilton, R., and M. Ackermann, "IPv6 Performance and Diagnostic Metrics (PDM) Destination Option", [RFC 8250](#), DOI 10.17487/RFC8250, September 2017, <<https://www.rfc-editor.org/info/rfc8250>>.
- [RFC8289] Nichols, K., Jacobson, V., McGregor, A., Ed., and J. Iyengar, Ed., "Controlled Delay Active Queue Management", [RFC 8289](#), DOI 10.17487/RFC8289, January 2018, <<https://www.rfc-editor.org/info/rfc8289>>.
- [RFC8290] Hoeiland-Joergensen, T., McKenney, P., Taht, D., Gettys, J., and E. Dumazet, "The Flow Queue CoDel Packet Scheduler and Active Queue Management Algorithm", [RFC 8290](#), DOI 10.17487/RFC8290, January 2018, <<https://www.rfc-editor.org/info/rfc8290>>.
- [RFC8404] Moriarty, K., Ed. and A. Morton, Ed., "Effects of Pervasive Encryption on Operators", [RFC 8404](#), DOI 10.17487/RFC8404, July 2018, <<https://www.rfc-editor.org/info/rfc8404>>.
- [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>>.

Appendix A. Revision information

-00 This is an individual draft for the IETF community.

-01 This draft was a result of walking away from the text for a few days and then reorganising the content.

-02 This draft fixes textual errors.

-03 This draft follows feedback from people reading this draft.

-04 This adds an additional contributor and includes significant reworking to ready this for review by the wider IETF community Colin Perkins joined the author list.

Comments from the community are welcome on the text and recommendations.

-05 Corrections received and helpful inputs from Mohamed Boucadair.

-06 Updated following comments from Stephen Farrell, and feedback via email. Added a draft conclusion section to sketch some strawman scenarios that could emerge.

-07 Updated following comments from Al Morton, Chris Seal, and other feedback via email.

-08 Updated to address comments sent to the TSVWG mailing list by Kathleen Moriarty (on 08/05/2018 and 17/05/2018), Joe Touch on 11/05/2018, and Spencer Dawkins.

-09 Updated security considerations.

-10 Updated references, split the Introduction, and added a paragraph giving some examples of why ossification has been an issue.

-01 This resolved some reference issues. Updated section on observation by devices on the path.

-02 Comments received from Kyle Rose, Spencer Dawkins and Tom Herbert. The network-layer information has also been re-organised after comments at IETF-103.

-03 Added a section on header compression and rewriting of sections referring to RTP transport. This version contains author editorial work and removed duplicate section.

-04 Revised following SecDir Review

- o Added some text on TLS story (additional input sought on relevant considerations).
- o [Section 2](#), paragraph 8 - changed to be clearer, in particular, added "Encryption with secure key distribution prevents"
- o Flow label description rewritten based on PDS/BCP RFCs.
- o Clarify requirements from RFCs concerning the IPv6 flow label and highlight ways it can be used with encryption. ([section 3.1.3](#))
- o Add text on the explicit spin-bit work in the QUIC DT. Added greasing of spin-bit. ([Section 6.1](#))
- o Updated [section 6](#) and added more explanation of impact on operators.
- o Other comments addressed.

Authors' Addresses

Godred Fairhurst
University of Aberdeen
Department of Engineering
Fraser Noble Building
Aberdeen AB24 3UE
Scotland

E-Mail: gorry@erg.abdn.ac.uk
URI: <http://www.erg.abdn.ac.uk/>

Colin Perkins
University of Glasgow
School of Computing Science
Glasgow G12 8QQ
Scotland

E-Mail: csp@cspcrkins.org
URI: <https://cspcrkins.org//>

