TSVWG
Internet-Draft
Intended status: Informational
Expires: January 14, 2021

G. Fairhurst
University of Aberdeen
C. Perkins
University of Glasgow

July 13, 2020

Considerations around Transport Header Confidentiality, Network Operations, and the Evolution of Internet Transport Protocols

draft-ietf-tsvwg-transport-encrypt-16

Abstract

To protect user data and privacy, Internet transport protocols have supported payload encryption and authentication for some time. Such encryption and authentication is now also starting to be applied to the transport protocol headers. This helps avoid transport protocol ossification by middleboxes, while also protecting metadata about the communication. Current operational practice in some networks inspect transport header information within the network, but this is no longer possible when those transport headers are encrypted.

This document discusses the possible impact when network traffic uses a protocol with an encrypted transport header. It suggests issues to consider when designing new transport protocols or features. These considerations arise from concerns such as network operations, prevention of network ossification, enabling transport protocol evolution and respect for user privacy.

Status of This Memo

This Internet-Draft will expire on January 14, 2021.

Copyright Notice

Table of Contents

1. Introduction

Transport protocols have supported end-to-end encryption of payload data for many years. Examples include Transport Layer Security (TLS) over TCP [RFC8446], Datagram TLS [RFC6347][I-D.ietf-tls-dtls13], Secure Real-time Transport Protocol (SRTP) [RFC3711], and tcpcrypt [RFC8548]. Some of these specifications also provide integrity protection of all, or part, of the transport header.

This end-to-end transport payload encryption brings many benefits in terms of providing confidentiality and protecting user privacy. The benefits have been widely discussed, for example in [RFC7624]. This document supports and encourages increased use of end-to-end payload encryption in transport protocols. The implications of protecting the transport payload data are therefore not further discussed in this document.

A further level of protection can be achieved by encrypting the entire network layer payload, including both the transport headers and the transport payload data. This does not expose any transport header information to devices in the network, and therefore also prevents modification along a network path. An example of encryption at the network layer is the IPsec Encapsulating Security Payload (ESP) [RFC4303] in tunnel mode. Virtual Private Networks (VPNs) typically also operate in this way. This form of encryption is not further discussed in this document.

There is also a middle ground, comprising transport protocols that encrypt some, or all, of the transport layer header information, in addition to encrypting the transport payload data. An example of such a protocol, that is now seeing widespread interest and deployment, is the QUIC transport protocol [I-D.ietf-quic-transport]. The encryption and authentication of transport header information can prevent unwanted modification of transport header information by network devices, reducing the risk of protocol ossification. It also reduces the amount of metadata about the progress of the transport connection that is visible to the network [RFC8558].

In this document, the term "transport header information" is used to describe transport layer information concerning the operation of the transport protocol (i.e., information used by the transport protocol that might be carried in a protocol header). This does not refer to transport payload data (i.e., information transferred by the transport service), which itself could be encrypted.

The direction in which the use of transport header encryption evolves could have significant implications on the way the Internet architecture develops, and therefore needs to be considered as a part of protocol design and evolution. This includes considering whether the endpoints permit (or are able to permit) network devices to observe specific information by explicitly exposing a transport header field (or a field derived from transport header information) to the network; whether it is intended that a network device can modify a transport header field; and whether any modification along the network path can be detected by the receiving endpoint. This can require changes to network operations and other practises and could drive changes to the design of network measurement for research, operational, and standardisation purposes.

As discussed in [RFC7258], the IETF has concluded that Pervasive Monitoring (PM) is a technical attack that needs to be mitigated in the design of IETF protocols, but RFC7258 also notes that "Making networks unmanageable to mitigate PM is not an acceptable outcome, but ignoring PM would go against the consensus documented here. An appropriate balance will emerge over time as real instances of this tension are considered". In support of achieving that balance, this document discusses design and deployment considerations for use of transport header encryption to protect against pervasive monitoring.

The transport protocols developed for the Internet are used across a wide range of paths across network segments with many different regulatory, commercial, and engineering considerations. This document considers some of the costs and changes to network management and research that are implied by widespread use of transport protocols that encrypt their transport header information. It reviews the implications of developing transport protocols that use end-to-end encryption to provide confidentiality of their transport layer headers, and considers the effect of such changes on transport protocol design, transport protocol evolution, and network operations. It also considers some anticipated implications on application evolution. This provides considerations relating to the design of transport protocols and features where the transport protocol encrypts some or all of their header information.

2. Context and Rationale

The transport layer provides end-to-end interactions between endpoints (processes) using a network path. Transport protocols layer over the network-layer service, and are usually sent in the payload of network-layer packets. Transport protocols support end-to-end communication between applications, using higher-layer protocols running on the end systems (i.e., transport endpoints).

This simple architectural view does not present one of the core functions of an Internet transport: to discover and adapt to the characteristics of the network 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.

Transport headers have end-to-end meaning, but have often been observed by equipment within the network. Transport protocol specifications have not tended to consider this. Designs have often failed to:

- specify what parts of the transport header can be modified by the network to signal to the transport, and in what way;

- indicate what parts of the transport header are intended to be invariant across protocol versions and visible to the network;

- indicate what parts of the transport header are intended expected to change in future and might need to be protected to prevent protocol ossification;

- and have often not defined which parts of the header need to be protected for privacy.

This motivates a need to change the way transport protocols are designed, modified, and specified.

Increasing concern about pervasive network monitoring [RFC7258][RFC7624], and growing awareness of the problem of protocol ossification caused by middlebox interference with Internet traffic, has motivated a shift in transport protocol design. For example, transport protocols, such as QUIC [I-D.ietf-quic-transport], encrypt the majority of their transport headers to prevent observation and protect against modification by the network, and to make explicit their invariants and what is intended to be visible to the network.

Transport header encryption is expected to form a core part of future transport protocol designs. It can help to protect against pervasive monitoring, improve privacy, and reduce protocol ossification. Transport protocols that use header encryption with secure key distribution can provide confidentiality and protection for some, or all, of the transport header, controlling what is visible to, and can be modified by the network.

The increased use of transport header encryption has benefits, but also has implications for the broader ecosystem. The transport community has, to date, relied on measurements and insights from the network operations community to understand protocol behaviour, and to inform the selection of appropriate mechanisms to ensure a safe, reliable, and robust Internet. In turn, network operators and access providers have relied upon being able to observe traffic patterns and requirements, both in aggregate and at the flow level, to help understand and optimise the behaviour of their networks.

Transport header encryption can be used to intentionally limit the information available to network observers. The widespread use would therefore limit such observations, unless transport protocols are modified to selectively expose transport header information outside of the encrypted transport header.

It is important to understand how transport header information is used by networks, to allow future protocol designs to make an informed choice on what, if any, transport layer information to expose to the network.

2.1. Use of Transport Header Information in the Network

In-network measurement of transport flow characteristics can be used to enhance performance, control cost and improve service reliability. To support network operations and enhance performance, some operators have deployed functionality that utilises on-path observations of the transport headers of packets passing through their network ([RFC8517] gives an operator perspective on such use).

When network devices rely on the presence of a header field or the semantics of specific header information, this can lead to ossification where an endpoint has to supply a specific header to receive the network service that it desires.

In some cases, network-layer use of transport layer information can be benign or advantageous to the protocol (e.g., recognising the start of a TCP connection, providing header compression for an SRTP flow [RFC3711], or explicitly using exposed protocol information to provide consistent decisions by on-path devices). Header compression (e.g., [RFC5795]) depends on understanding of transport header and the way fields change packet-by-packet; as also do techniques to improve TCP performance by transparent modification of acknowledgement traffic [RFC3449]. Introducing a new transport protocol or changes to existing transport header information prevent these methods being used or require the network devices to be updated.

However, in other cases, ossification can have unwanted outcomes. Ossification can frustrate the evolution of a transport protocol. A mechanism implemented in a network device, such as a firewall, that requires a header field to have only a specific known set of values can prevent the device from forwarding packets using a different version of the protocol that introduces a feature that changes to a new value for the observed field.

An example of this type ossification was observed in the development of TLS 1.3 [RFC8446], where the design needed to function in the presence of deployed middleboxes that relied on the presence of certain header fields exposed in TLS 1.2 [RFC5426]. The design of Multipath TCP (MPTCP) [RFC8684] also had to be revised to account for middleboxes (known as "TCP Normalizers") that monitor the evolution of the window advertised in the TCP header and then reset connections when the window did not grow as expected. Similarly, issues have been reported using TCP. For example, TCP Fast Open [RFC7413] can experience middleboxes that modify the transport header of packets by removing "unknown" TCP options, segments with unrecognised TCP options can be dropped, segments that contain data and set the SYN bit can be dropped, or middleboxes that disrupt connections that send data before completion of the three-way handshake.

Other examples of ossification have included middleboxes that modify transport headers by rewriting TCP sequence and acknowledgement numbers, but are unaware of the (newer) TCP selective acknowledgement (SACK) option and therefore fail to correctly rewrite the SACK information to match the changes that were made to the fixed TCP header, preventing SACK from operating correctly.

In all these cases, middleboxes with a hard-coded, but incomplete, understanding of transport behaviour, interacted poorly with transport protocols after the transport behaviour was changed. In some case, the middleboxes modified or replaced information in the transport protocol header.

Transport header encryption prevents an on-path device from observing the transport headers, and therefore stops mechanisms being built that directly rely on or infer semantics of the transport header information. Encryption is normally combined with authentication of the protected information. RFC 8546 summarises this approach, stating that it is "The wire image, not the protocol’s specification, determines how third parties on the network paths among protocol participants will interact with that protocol" (Section 1 of

[RFC8546]), and it can be expected that header information that is not encrypted will become ossified.

While encryption can reduce ossification of the transport protocol, it does not itself prevent ossification of the network service. People seeking to understand network traffic could still come to rely on pattern inferences and other heuristics or machine learning to derive measurement data and as the basis for network forwarding decisions [RFC8546]. This can also create dependencies on the transport protocol, or the patterns of traffic it can generate, also in time resulting in ossification of the service.

2.2. Authentication of Transport Header Information

The designers of a transport protocol have to decide whether to encrypt all, or a part of, the transport layer information. Section 4 of [RFC8558] states: "Anything exposed to the path should be done with the intent that it be used by the network elements on the path".

Protocol designers can decide not to encrypt certain transport header fields, making those fields observable in the network, or can define new fields designed to explicitly expose observable transport layer information to the network. Where exposed fields are intended to be immutable (i.e., can be observed, but not modified by a network device), the endpoints are encouraged to use authentication to provide a cryptographic integrity check that can detect if these immutable fields have been modified by network devices. Authentication can also help to prevent attacks that rely on sending packets that fake exposed control signals in transport headers (e.g., TCP RST spoofing).

Making a part of a transport header observable or exposing new header fields can lead to ossification of that part of a header as network devices come to rely on observations of the exposed fields. A protocol design that provides an observable field might want to restrict the choice of usable values in a field by intentionally varying the format and/or value of the field to reduce the chance of ossification (see Section 4).

2.3. Perspectives on Observable Transport Header Fields

Transport header fields have been observed within the network for a variety of purposes. Some purposes are related to network management and operations. Use cases where the network devices intentionally modify mutable transport layer information are out of scope and are not described further in this document. More information may be found in other RFCs (e.g., [RFC3449], [RFC3135], [RFC8404], [RFC8462]), and [RFC8517].

The list below provides an overview with different uses of exposed immutable information.

Network Operations: A transport protocol with observable header information can enable explicit measurement and analysis of protocol performance, network anomalies, and failure pathologies at any point along the Internet path. In many cases, it is important to relate observations to specific equipment/configurations, to a specific network segment, or sometimes to a specific protocol or application.

When transport header information is not observable, it cannot be used by network operators. Some operators might work without that information, or some might turn to more ambitious ways to collect, estimate, or infer this data. (Operational practises aimed at guessing transport parameters are out of scope for this document, and are only mentioned here to recognise that encryption does not stop operators from attempting to apply practises that have been used with unencrypted transport headers.)

See also Section 3, Section 5, Section 7.4 and Section 7.5.

Analysis of Aggregate Traffic: Observable transport headers have been utilised to determine which transport protocols and features are being used across a network segment, and to measure trends in the pattern of usage. For some use cases, end-to-end measurements/traces are sufficient and can assist in developing and debugging new transports and analysing their deployment. In other uses, it is important to relate observations to specific equipment/configurations or particular network segments.

This information can help anticipate the demand for network upgrades and roll-out, or affect on-going traffic engineering activities performed by operators such as determining which parts of the path contribute delay, jitter, or loss.

Tools that rely upon observing specific headers, could fail to produce useful data when those headers are encrypted. While this impact could, in many cases, be small, there are scenarios where operators have actively monitored and supported particular services, e.g., to explore issues relating to Quality of Service (QoS), to perform fast re-routing of critical traffic, to mitigate the characteristics of specific radio links, and so on.

Troubleshooting: Observable transport headers have been utilised by operators as a part of network troubleshooting and diagnostics. Metrics derived from this observed header information can help localise the network segment introducing the loss or latency. Effective troubleshooting often requires understanding of transport behaviour. Flows experiencing packet loss or jitter are hard to distinguish from unaffected flows when only observing network layer headers.

Observable transport feedback information (e.g., RTP Control Protocol (RTCP) reception reports [RFC3550]) can explicitly make loss metrics visible to operators. Loss metrics can also be deduced with more complexity from other header information (e.g., by observing TCP SACK blocks). When the transport header information is encrypted, explicit observable fields could also be made available at the network or transport layers to provide these functions. [RFC8558] motivates the design of signals to focus on their usage, decoupled from the internal design of the protocol state machine. This could avoid ossifying the protocol around the design of a specific protocol mechanism.

See also Section 3.4 and Section 5.

Network Protection: Observable transport headers currently provide information that is useful input to classify and detect anomalous events, such as changes in application behaviour or distributed DoS attacks. Operators often seek to uniquely disambiguate unwanted traffic.

Where flows cannot be disambiguated based on transport header information, this could result in less-efficient identification of unwanted traffic, the introduction of rate limits for uncharacterised traffic, or the use of heuristics to identify anomalous flows.

See also Section 7.2 and Section 7.3.

Verifiable Data: Observable transport headers can be used to provide open and verifiable measurements to support operations, research, and protocol development. The ability of multiple stake holders to review transport header traces helps develop insight into performance and traffic contribution of specific variants of a protocol. Independently observed data is important to help ensure the health of the research and development communities.

When transport header information can not be observed, this can reduce the range of actors that can observe data. This limits the information sources available to the Internet community to understand the operation of transport protocols, reducing information to inform design decisions and standardisation of the new protocols/features and related operational practises

See also Section 7.

SLA Compliance: Observable transport headers coupled with published transport specifications allow operators and regulators to explore the compliance with Service Level Agreements (SLAs).

When transport header information can not be observed, other methods have to be found to confirm that the traffic produced conforms to the expectations of the operator or developer.

Independently verifiable performance metrics can be utilised to demonstrate regulatory compliance in some jurisdictions, and as a basis for informing design decisions. This can bring assurance to those operating networks, often avoiding deployment of complex techniques that routinely monitor and manage Internet traffic flows (e.g., avoiding the capital and operational costs of deploying flow rate-limiting and network circuit-breaker methods [RFC8084]).

See also Section 5 and Section 7.1 to Section 7.4.

This analysis does not judge whether specific practises are necessary. It is not an endorsement of any particular practice.

3. Current uses of Transport Headers within the Network

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 can improve privacy, and can help to mitigate certain attacks, but can also affect network operations [RFC8404].

When considering what parts of the transport headers should be encrypted to provide confidentiality, and what parts should be visible to the network (including non-encrypted but authenticated headers), it is necessary to consider both the impact on network operations and management, and the implications for ossification and user privacy [Measurement]. Different parties will view the relative importance of these concerns differently. For some, the benefits of encrypting all the transport headers outweigh the impact of doing so; others might analyse the security, privacy, and ossification impacts and arrive at a different trade-off.

This section reviews examples of the observation of transport layer headers within the network. Unencrypted transport headers provide information can support network operations and management, and this section notes some ways in which this has been done. Unencrypted transport header information also contributes metadata that can be exploited for purposes unrelated to network transport measurement, diagnostics or troubleshooting (e.g., to block or to throttle traffic from a specific content provider), and this section also notes some threats relating to unencrypted transport headers.

Exposed transport information also provides a source of information that contributes to linked data sets, which could be exploited to deduce private information, e.g., user patterns, user location, tracking behaviour, etc. This might reveal information the parties did not intend to be revealed. [RFC6973] aims to make designers, implementers, and users of Internet protocols aware of privacy-related design choices in IETF protocols.

This section does not consider intentional modification of transport headers by middleboxes, such as in Network Address Translation (NAT) or Firewalls. Common issues concerning IP address sharing are described in [RFC6269].

3.1. To Identify Transport Protocols and Flows

Information in exposed transport layer headers can be used by the network to identify transport protocols and flows [RFC8558]. The ability to identify transport protocols, flows, and sessions is a common function performed, for example, by measurement activities, QoS classifiers, and firewalls. These functions can be beneficial, and performed with the consent of, and in support of, the end user. Alternatively, a network operator could use the same mechanisms to support practises that are adversarial to the end user, including blocking, de-prioritising, and monitoring traffic without consent.

Observable transport header information, together with information in the network header, has been used to identify flows and their connection state, together with the set of protocol options being used. 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. They also have the possibility to negotiate additional headers at connection setup, identified by an option number in the transport header.

In some uses, an assigned transport port (e.g., 0..49151) can identify the upper-layer protocol or service [RFC7605]. However, port information alone is not sufficient to guarantee identification. Applications can use arbitrary ports and do not need to use assigned port numbers. The use of an assigned port number is also not limited to the protocol for which the port is intended. Multiple sessions can also be multiplexed on a single port, and ports can be re-used by subsequent sessions.

Some flows can 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].

When transport header information cannot be observed, this removes information that could have been used to classify flows by passive observers along the path. More ambitious ways could be used to collect, estimate, or infer flow information, including heuristics based on the analysis of traffic patterns. For example, an operator that cannot access the Session Description Protocol (SDP) session descriptions [RFC4566] to classify a flow as audio traffic, might instead use (possibly less-reliable) heuristics to infer that short

UDP packets with regular spacing carry audio traffic. Operational practises aimed at inferring transport parameters are out of scope for this document, and are only mentioned here to recognise that encryption does not prevent operators from attempting to apply practises that were used with unencrypted transport headers.

The IAB [RFC8546] have provided a summary of expected implications of increased encryption on network functions that use the observable headers and describe the expected benefits of designs that explicitly declare protocol invariant header information that can be used for this purpose.

3.2. To Understand Transport Protocol Performance

Information in exposed transport layer headers can be used by the network to understand transport protocol performance and behaviour.

3.2.1. Using Information Derived from Transport Layer Headers

Observable transport headers enable explicit measurement and analysis of protocol performance, network anomalies, and failure pathologies at any point along the Internet path. Some operators use passive monitoring to manage their portion of the Internet by characterising the performance of link/network segments. Inferences from transport headers are used to derive performance metrics. A variety of open source and commercial tools have been deployed that utilise transport header information in this way to derive the following metrics:

Traffic Rate and Volume: Volume measures per-application can be used to characterise the traffic that uses a network segment or the pattern of network usage. Observing the protocol sequence number and packet size offers one way to measure this (e.g., measurements observing counters in periodic reports such as RTCP; or measurements observing protocol sequence numbers in statistical samples of packet flows, or specific control packets, such as those observed at the start and end of a flow).

Measurements can be per endpoint, or for an endpoint aggregate. This can be used, for example, to assess subscriber usage or for billing purposes.

Measurements can also be used to trigger traffic shaping, and to associate QoS support within the network and lower layers. This can be done with consent and in support of an end user, to improve quality of service; or can be used by the network to de-prioritise certain flows without user consent.

Volume measures can also be valuable for capacity planning and providing detail of trends in usage.

The traffic rate and volume can be determined providing that the packets belonging to individual flows can be identified, but there might be no additional information about a flow when the transport headers cannot be observed.

Loss Rate and Loss Pattern: Flow loss rate can be derived (e.g., from transport sequence numbers or inferred from observing transport protocol interactions) 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., due to interference on a radio link), buffering loss (e.g., overflow due to congestion, Active Queue Management (AQM) [RFC7567], or inadequate provision following traffic pre-emption), and policing (traffic management) [RFC2475]. Understanding flow loss rates requires either observing sequence numbers in network or transport headers, or maintaining per-flow packet counters (flow identification often requires transport layer information). Per-hop loss can also sometimes be monitored at the interface level by devices in the network.

Losses can often occur as bursts, randomly-timed events, etc. The pattern of loss can provide insight into the cause of loss. It can also be valuable to understand the conditions under which loss occurs, which usually requires relating loss to the traffic flowing at a network node or segment at the time of loss. This can also help identify cases where loss could have been wrongly identified, or where the transport did not require transmission of a lost packet.

Throughput and Goodput: Throughput is the amount of payload data sent by a flow per time interval. Goodput (see Section 2.5 of [RFC7928]) is a measure of useful data exchanged (the ratio of useful data to total volume of traffic sent by a flow). The throughput of a flow can be determined in the absence of transport header information, providing that the individual flow can be identified, and the overhead known. Goodput requires ability to differentiate loss and retransmission of packets, for example by observing packet sequence numbers in the TCP or RTP headers [RFC3550].

Latency: Latency is a key performance metric that impacts application and user-perceived response times. It often indirectly impacts throughput and flow completion time. This 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 queueing 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 [RFC7799] 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 allows measurement of 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 configuration changes and to 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 often require tuning [RFC8290] [RFC8289] [RFC8033] because they cannot scale across all possible deployment scenarios.

Latency and round-trip time information can potentially expose some information useful for approximate geolocation, as discussed in [PAM-RTT]. Encrypting transport headers can reduce the latency information that is available.

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. For example, jitter metrics are often cited when characterising paths supporting real-time traffic. The expected performance of such applications, can be inferred from a 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. Network tools can detect and measure unwanted/excessive reordering, and the impact on transport performance.

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 potential 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] [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 assist in understanding the context under which the data was collected, including the time, observation point [RFC7799], 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 has to trust the source of this data and the method used to generate the summary information.

These metrics can support network operations, inform capacity planning, and assist in determining the demand for equipment and/or configuration changes by network operators. They can also inform Internet engineering activities by informing the development of new protocols, methodologies, and procedures.

In some cases, measurements could involve active injection of test traffic to perform a measurement (see Section 3.4 of [RFC7799]). 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 cost in running such tests (e.g., the implications of capacity tests in a mobile network are obvious). Some active measurements [RFC7799] (e.g., response under load or particular workloads) perturb other traffic, and could require dedicated access to the network segment.

Passive measurements (see Section 3.6 of [RFC7799]) 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 3.3.1). Measurements can rely on observing packet headers, which is not possible if those headers are encrypted, but could utilise information about traffic volumes or patterns of interaction to deduce metrics.

One alternative approach is to use in-network techniques, which does not require the cooperation of an endpoint (see Section 5).

3.2.2. Using Information Derived from Network Layer Header Fields

Information from the transport header 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, or by firewalls to implement access rules (see also Section 2.2.2 of [RFC8404]). Operators can use such policies to support user applications and to protect against unwanted traffic. Such policies can also be used without user consent, to de-prioritise certain applications or services, for example.

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 header information. Traffic that cannot be classified typically receives a default treatment. Some networks block or rate traffic that cannot be classified.

Transport layer information can also be explicitly carried in network-layer header fields that are not encrypted, serving as a replacement/addition to the exposed transport header information [RFC8558]. This information can enable a different forwarding treatment by the network, even when a transport employs encryption to protect other header information.

The user of a transport that multiplexes multiple sub-flows might want to obscure 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 service requirements of individual sub-flows. There are several ways this could be done:

IP Address: Applications normally 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 affect 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 sub-flows. 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".

Once set, a flow label can provide information that can help inform network-layer queueing and forwarding [RFC6438], for example with Equal Cost Multi-Path routing and Link Aggregation [RFC6294]. Considerations when using IPsec are further described in [RFC6438].

The choice of how to assign a flow label needs to avoid introducing linkability that a network device could observe. Inappropriate use by the transport can have privacy implications (e.g., assigning the same label to two independent flows that ought not to be classified the same).

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 queueing and forwarding, rather than an operator inferring traffic requirements from transport and application headers via a multi-field classifier. Inappropriate use by the transport can have privacy implications (e.g., assigning a different DSCP to a subflow could assist in a network device discovering the traffic pattern used by an application, assigning the same label to two independent flows that ought not to be classified the same). The field is mutable, i.e., some network devices can be expected to change this field (use of each DSCP value is defined by an RFC).

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

Using Explicit Congestion Marking: ECN [RFC3168] is a transport mechanism that uses 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 flow. An ECN-capable transport can offer benefits when used over a path with equipment that implements an AQM method with 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 have to be available to network operators and researchers to understand the implication of configuration choices and transport behaviour as the use of ECN increases and new methods emerge [RFC7567].

Network-Layer Options Network protocols can carry optional headers. These can be used to explicitly expose transport header information to on-path devices operating at the network layer (as discussed further in Section 5).

IPv4 [RFC0791] has provision for optional header fields identified by an option type field. IP routers can examine these headers and are required to ignore IPv4 options that they does not recognise. Many current paths include network devices that forward packets that carry options on a slower processing path. Some network devices (e.g., firewalls) can be (and are) configured to drop these packets [RFC7126]. BCP 186 [RFC7126] provides Best Current Practice guidance on how operators should treat IPv4 packets that specify options.

IPv6 can encode optional network-layer information in separate headers that may be placed between the IPv6 header and the upper-layer header [RFC8200]. The Hop-by-Hop options header, when present, immediately follows the IPv6 header. IPv6 permits this header to be examined by any node along the path. While [RFC7872] required all nodes to examine and process the Hop-by-Hop options header, it is now expected [RFC8200] that nodes along a path only examine and process the Hop-by-Hop options header if explicitly configured to do so.

When transport headers cannot be observed, operators are unable to use this information directly. Careful use of the network layer features can help provide similar information in the case where the network is unable to inspect transport protocol headers. Section 6 describes use of network extension headers.

3.3. To Support Network Operations

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 transport protocols performing header encryption, with the obvious exception of VPNs and IPsec.

When encryption hides more layers in each packet, people seeking understanding of the network operation rely more on pattern inference and other heuristics. 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 necessary to interpret this data to answer questions about network performance or understand a pathology. Collecting and coordinating such metadata is more difficult when the observation point is at a different location to the bottleneck/device under evaluation [RFC7799].

Packet sampling techniques are 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, but 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.

3.3.1. Problem Location

In network measurements of transport header information can be used to locate 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 in network observation points have to be used. 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.

3.3.2. Network Planning and Provisioning

Traffic rate and volume measurements are used by operators to help plan deployment of new equipment and configuration 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.

Trends in aggregate traffic can be observed and can be related to the endpoint addresses being used, but when transport header information is not observable, it might be impossible to correlate patterns in measurements with changes in transport protocols. This increases the dependency on other indirect sources of information to inform planning and provisioning.

3.3.3. Service Performance Measurement

Performance measurements (e.g., throughput, loss, latency) can be used by various actors to analyse the service offered to the users of a network segment, and to inform operational practice.

3.3.4. Compliance with Congestion Control

The traffic that can be observed by on-path network devices (the "wire image") 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.

Operators can manage traffic load (e.g., when the network is severely overloaded) by deploying rate-limiters, traffic shaping, or network transport circuit breakers [RFC8084]. The information provided by observing transport headers is a source of data that can help to inform such mechanisms.

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 the shared Internet. TCP algorithms have been continuously improved over decades, and have reached a level of efficiency and correctness that is difficult to match in custom application-layer mechanisms [RFC8085].

A standards-compliant TCP stack provides congestion control that is 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 transports (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 an 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 under 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 and flows that contribute to persistent congestion is important to the safe operation of network infrastructure, and 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 have to employ mechanisms to prevent collapse, avoid unacceptable contributions to jitter/latency, and to establish an acceptable share of capacity with concurrent traffic [RFC8085].

A network operator can observe the headers of transport protocols layered above UDP to understand if the datagram flows comply with congestion control expectations. This can help inform a decision on whether it might be appropriate to deploy methods such as rate-limiters to enforce acceptable usage.

UDP flows that expose a well-known header can 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 RTP header information and RTCP reports for real-time flows (see Section 3.2). The Secure RTP and RTCP extensions [RFC3711] were explicitly designed to expose some header information to enable such observation, while protecting the payload data.

3.4. To Support Network Diagnostics and Troubleshooting

Transport header information can be utilised for a variety of operational tasks [RFC8404]: to diagnose network problems, assess network provider performance, evaluate equipment or protocol performance, capacity planning, management of security threats (including DoS), and responding to user performance questions. Section 3.1.2 and Section 5 of [RFC8404] provide further examples.

Operators can monitor the health of a portion of the Internet, to provide early warning and trigger action. Traffic and performance measurements can assist in setting buffer sizes, 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 other cases, measurement involves dissecting network traffic flows. Observed transport header information can help identify whether link/

network tuning is effective and alert to potential problems that can be hard to derive from link or device measurements alone.

An alternative could rely on access to endpoint diagnostic tools or user involvement in diagnosing and troubleshooting unusual use cases or to troubleshoot non-trivial problems.

Another approach is to use traffic pattern analysis. Such tools can provide useful information during network anomalies (e.g., detecting significant reordering, high or intermittent loss), however indirect measurements would need to be carefully designed to provide information for diagnostics and troubleshooting.

The design trade-offs for radio networks are often very different from those of wired networks. A radio-based network (e.g., cellular mobile, enterprise Wireless LAN (WLAN), 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. For radio links, the use 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.

Lack of tools and resulting information can reduce 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 respond to other queries about the network service.

A network operator supporting traffic that uses transport header encryption is unable to use tools that rely on transport protocol information. However, the use of encryption has the desirable effect of preventing unintended observation of the payload data and these tools seldom seek to observe the payload, or other application details. A flow that hides its transport header information could imply "don't touch" to some operators. This might limit a trouble-shooting response to "can't help, no trouble found".

3.5. To Support Header Compression

Header compression saves link capacity 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. Examples of header compression include use with TCP/IP and RTP/UDP/IP flows [RFC2507], [RFC2508], [RFC4995].

While it is possible to compress only the network layer headers, significant savings can be made if both the network and transport layer headers are compressed together as a single unit. The SRTP 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 a fall back to compressing only the network layer headers, with a significant reduction in efficiency.

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 (e.g., using TLS). This can hide information from an eavesdropper in the network. It can also help protect the privacy of a user, by hiding data relating to user/device identity or location.

4.1. Motivation

There are several motivations for encryption:

- One motive to encrypt transport headers is in response to a growing awareness of the implications of network ossification from network devices that inspect transport headers. Once a network devices observes a transport header and becomes reliant upon using it, the overall use of that field can become ossified, preventing new protocols and mechanisms from being deployed. One of the benefits of encrypting transport headers is that it can help improve the pace of transport development by eliminating interference by deployed middleboxes.

- Another motivation stems from increased concerns about privacy and surveillance. Users value the ability to protect their identity and location, and defend against analysis of the traffic. Revelations about the use of pervasive surveillance [RFC7624] have, to some extent, eroded trust in the service offered by network operators and have led to an increased use of encryption to avoid unwanted eavesdropping on communications. Concerns have also been voiced about the addition of information to packets by third parties to provide analytics, customisation, advertising, cross-site tracking of users, to bill the customer, or to selectively allow or block content. Whatever the reasons, the IETF is designing protocols that include transport header encryption (e.g., QUIC [I-D.ietf-quic-transport]) to supplement the already widespread payload encryption, and to further limit exposure of transport metadata to the network.

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

- On the one hand, future Internet protocols that support transport header 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, and can improve privacy by reducing leakage of transport metadata.
- On the other hand, encryption of transport layer 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.

A decision to use transport header encryption can improve user privacy, and can reduce protocol ossification and help the evolution of the transport protocol stack, but is also has implications for network operations and management.

4.2. Approaches to Transport Header Protection

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, allows in-network devices to observe these fields. An integrity check is not able to 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. Such authentication might 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. The IPsec Encapsulating Security Payload (ESP) [RFC4303] can also provide authentication and integrity without confidentiality using the NULL encryption algorithm [RFC2410]. SRTP [RFC3711] is another example of a transport protocol that allows header authentication.

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 (explicitly exposed either in a transport header field or lower layer protocol header). A design that only exposes immutable fields can also perform end-to-end authentication of these fields across the path to prevent undetected modification of the immutable transport headers.

Mutable fields in the transport header provide opportunities where network devices can modify the transport behaviour (e.g., the extended headers described in [I-D.trammell-plus-abstract-mech]).

An example of a method that encrypts some, but not all, transport header 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 have to be made 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.

Greasing: 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. This seeks 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. It is not a security mechanism, although use can be combined with an authentication mechanism.

As seen, different transports use encryption to protect their header information to varying degrees. The trend is towards increased protection.

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

An on-path device can make measurements by utilising additional protocol headers carrying operations, administration and management (OAM) information in an additional packet header. Using network-layer approaches to reveal information has the potential that the same method (and hence same observation and analysis tools) can be consistently used by multiple transport protocols. This approach also could be applied to methods beyond OAM (see Section 6). There can also be less desirable implications from separating the operation of the transport protocol from the measurement framework.

5.1. Use of OAM within a Maintenance Domain

OAM information can be added 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], or as a part of encapsulation protocol). The additional header information is typically removed the at the egress of the maintenance domain.

Although some types of measurements are supported, this approach 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 appropriate 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.

5.2. Use of OAM across Multiple Maintenance Domains

OAM information can also be added at the network layer as an IPv6 extension header or an IPv4 option. This information can be used across multiple network segments, or between the transport endpoints.

One example is the IPv6 Performance and Diagnostic Metrics (PDM) destination option [RFC8250]. This allows a sender to optionally include a destination option that caries 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 has to be explicitly enabled at the sender.

6. Intentionally Exposing Transport Information to the Network

A transport protocol can choose to expose certain transport information to on-path devices operating at the network layer by sending observable fields. One approach is to make an explicit choice not to encrypt certain transport header fields, making this transport information observable by the network. Another approach is to choose to expose transport information through the use of network-layer extension headers. Both are examples of explicit information intended to be used by network devices on the path [RFC8558].

Whatever the mechanism used to expose the information, a decision to only expose specific transport information, places the transport endpoint in control of what to expose or not to expose outside of the encrypted transport header. This decision can then be made independently of the transport protocol functionality. This can be done by exposing part of the transport header or as a network layer option/extension.

6.1. Exposing Transport Information in Extension Headers

At the network-layer, packets can carry optional headers (similar to Section 5) that may be used to explicitly expose transport header information to the on-path devices operating at the network layer (Section 3.2.2). For example, an endpoint that sends an IPv6 Hop-by-Hop option [RFC8200] can provide explicit transport layer information that can be observed and used by network devices on the path.

Network-layer optional headers explicitly indicate the information that is exposed, whereas use of exposed transport header information first requires an observer to identify the transport protocol and its format. See Section 3.1 for further discussion of transport protocol identification.

An arbitrary path can include one or more network devices that drop packets that include a specific header or option used for this purpose (see [RFC7872]). This could impact the proper functioning of the protocols using the path. Protocol methods can be designed to probe to discover whether the specific option(s) can be used along the current path, enabling use on arbitrary paths.

6.2. Common Exposed Transport Information

There are opportunities for multiple transport protocols to consistently supply common observable information [RFC8558]. A common approach can result in an open definition of the observable fields. This has the potential that the same information can be utilised across a range of operational and analysis tools.

6.3. Considerations for Exposing Transport Information

The motivation to reflect actual transport header information and the implications of network devices using this information has to be considered when proposing such a method. RFC8558 summarises this as "When signals from endpoints to the path are independent from the signals used by endpoints to manage the flow's state mechanics, they may be falsified by an endpoint without affecting the peer's understanding of the flow's state. For encrypted flows, this divergence is not detectable by on-path devices." [RFC8558].

Considerations concerning the selection of appropriate information to expose include:

- On the one hand, explicitly exposing derived fields containing relevant transport information (e.g., metrics for loss, latency, etc) can avoid network devices needing to derive this information from other header fields. This could result in development and evolution of transport-independent tools around a common observable header, and permit transport protocols to also evolve independently of this ossified header [RFC8558].
- On the other hand, protocols and implementations may not consistently expose external information that reflects the actual internal information used by the protocol itself. An endpoint/ protocol could choose to expose transport header information to optimise the benefit it gets from the network [RFC8558]. The value of this information would be enhanced if the exposed information could be verified to match the protocol's observed behavior.

7. Implications of Protecting the Transport Headers

The choice of which transport header fields to expose and which to encrypt is a design decision for the transport protocol. Selective encryption requires trading conflicting goals of observability and network support, privacy, and risk of ossification, to decide what header fields to protect and which to make visible.

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.

7.1. Independent Measurement

Independent observation by multiple actors is important if the transport community is to maintain an accurate understanding of the network. Encrypting transport header encryption changes the ability to collect and independently analyse data. Internet transport protocols employ a set of mechanisms. Some of these have to work in cooperation with the network layer for loss detection and recovery, congestion detection and control. Others have 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, many real-time applications use UDP, and much of this traffic uses 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 became 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 asses congestion and requests for retransmission) provide an incentive for the sending endpoint to provide correct information, since the protocol will not work otherwise. This increases 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 transport 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 or RTP implementation to put incorrect information in this transport header. A network device can have confidence that the well-known (and ossified) transport header information represents the actual state of the endpoints.

When encryption is used to hide some or all of the transport headers, the transport protocol chooses which 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 (or an existing transport modified), for example, to provide summary data regarding its performance, congestion control state, etc., or to make available explicit measurement information. For example, a QUIC endpoint can optionally set the spin bit to reflect to explicitly reveal the RTT of an encrypted transport session to the on-path network devices [I-D.ietf-quic-transport]).

When providing or using such information, it is 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 measurable information 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. For some operational uses, the information has to contain sufficient detail to understand, and possibly reconstruct, the network traffic pattern for further testing. In this case, operators have to gain the trust of transport protocol implementers if the transport headers are to correctly reveal such information.

OAM data records [I-D.ietf-ippm-ioam-data] could be embedded into a variety of encapsulation methods at different layers to support the goals of a specific operational domain. OAM-related metadata can support functions such as performance evaluation, path-tracing, path verification information, classification and a diversity of other uses. When encryption is used to hide some or all of the transport headers, analysis requires coordination between actors at different layers to successfully characterise flows and correlate the performance or behaviour of a specific mechanism with the configuration and traffic using operational equipment (e.g., combining transport and network measurements to explore congestion control dynamics, the implications of designs for active queue management or circuit breakers).

Some measurements could be completed by utilising endpoint-based logging (e.g., based on Quic-Trace [Quic-Trace]). Such information has a diversity of uses, including developers wishing to debug/ understand the transport/application protocols with which they work, researchers seeking to spot trends and anomalies, and to characterise variants of protocols. A standard format for endpoint logging could allow these to be shared (after appropriate anonymisation) to understand performance and pathologies. Measurements based on logging have to establish the validity and provenance of the logged information to establish how and when traces were captured.

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

7.2. 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 might not have a significant collateral impact on the performance of other traffic that shares this network segment. Once the proportion of this traffic increases, monitoring the traffic can determine if appropriate safety measures have 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 has to be considered in determining how this activity is performed. On a shorter timescale, information could also have to be collected to manage DoS attacks against the infrastructure.

7.3. 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 3.3.4). 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 could 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 has to be considered when evaluating the impact of designs for transport encryption [RFC5218].

7.4. Impact on Network Operations

Some network operators currently use observed transport header information as a part of their operational practice, and have developed tools and techniques that use information observed in currently deployed transports and their applications. A variety of open source and proprietary tools have been deployed that use this information for a variety of short and long term measurements. Encryption of the transport header information prevents tooling from directly observing the transport header information, limiting its utility.

Alternative diagnostic and troubleshooting tools would have to be developed and deployed is transport header encryption is widely deployed. Introducing a new protocol or application might then require these tool chains and practises to be updated, and could in turn impact operational mechanisms, and policies. Each change can introduce associated costs, including the cost of collecting data, and the tooling to handle multiple formats (possibly as these co-exist in the network, when measurements 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 overall operational impact 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.

7.5. Impact on Research, Development and Deployment

Transport protocol evolution, and the ability to measure and understand the impact of protocol changes, have to proceed hand-in-hand. A transport protocol that provides observable headers can be used to provide open and verifiable measurement data. Observation of pathologies has a critical role in the design of transport protocol mechanisms and development of new mechanisms and protocols. This helps 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. The ability of other stake holders to review transport header traces helps develop insight into performance and traffic contribution of specific variants of a protocol.

Development of new transport protocol mechanisms has to consider the scale of deployment and the range of environments in which the transport is used. Experience has shown that it is often difficult to correctly implement new mechanisms [RFC8085], and that mechanisms often evolve as a protocol matures, or in response to changes in network conditions, changes in network traffic, or changes to application usage. Analysis is especially valuable when based on the behaviour experienced across a range of topologies, vendor equipment, and traffic patterns.

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.

Encryption of transport header information could reduce the range of actors that can observe useful data. This would limit the information sources available to the Internet community to understand the operation of new transport protocols, reducing information to inform design decisions and standardisation of the new protocols and related operational practises. 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.

Independently observed data is also important to ensure the health of the research and development communities and can help promote acceptance of proposed specifications by the wider community (e.g., as a method to judge the safety for Internet deployment) and provides valuable input during standardisation. Open standards motivate a desire to include independent observation and evaluation of performance data, which in turn demands control/understanding about where and when measurement samples are collected. This requires consideration of the methods used to observe data and the appropriate balance between encrypting all and no transport header information.

8. Conclusions

Header encryption and strong integrity checks are being incorporated into new transport protocols and have important benefits. The pace of development of transports using the WebRTC data channel, and the rapid deployment of the 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.

This document has described some current practises, and the implications for some stake holders, when transport layer header encryption is used. It does not judge whether these practises are necessary, or endorse the use of any specific practise. Rather, the intent is to highlight operational tools and practises to consider when designing and modifying transport protocols, so protocol designers can make informed choice about what transport header fields to encrypt, and whether it might be beneficial to make an explicit choice to expose certain fields to the network. In making such a decision, it is important to balance:

- User Privacy: The less transport header information that is exposed to the network, the lower the risk of leaking metadata that might have privacy implications for the users. Transports that chose to expose some header fields need to make a privacy assessment to understand the privacy cost versus benefit trade-off in making that information available. The process used to define and expose the QUIC spin bit to the network is an example of such an analysis.

- Protocol Ossification: Unencrypted transport header fields are likely to ossify rapidly, as network devices come to rely on their presence, making it difficult to change the transport in future. This argues that the choice to expose information to the network is made deliberately and with care, since it is essentially defining a stable interface between the transport and the network. Some protocols will want to make that interface as limited as possible; other protocols might find value in exposing certain information to signal to the network, or in allowing the network to change certain header fields as signals to the transport. The visible wire image of a protocol should be explicitly designed.

- Impact on Operational Practice: The network operations community has long relied on being able to understand Internet traffic patterns, both in aggregate and at the flow level, to support network management, traffic engineering, and troubleshooting. Operational practice has developed based on the information available from unencrypted transport headers. The IETF has supported this practice by developing operations and management specifications, interface specifications, and associated Best Current Practises. Widespread deployment of transport protocols that encrypt their information might impact network operations, unless operators can develop alternative practises that work without access to the transport header.

- Pace of Evolution: Removing obstacles to change can enable an increased pace of evolution. If a protocol changes its transport header format (wire image) or their transport behaviour, this can result in the currently deployed tools and methods becoming no longer relevant. Where this needs to be accompanied by development of appropriate operational support functions and procedures, it can incur a cost in new tooling to catch-up with each change. Protocols that consistently expose observable data do not require such development, but can suffer from ossification and need to consider if the exposed protocol metadata has privacy implications, There is no single deployment context, and therefore designers need to consider the diversity of operational networks (ISPs, enterprises, Distributed DoS (DDoS) mitigation and firewall maintainers, etc.).

- Supporting Common Specifications: Common, open, specifications can stimulate engagement by developers, users, researchers, and the broader community. Increased protocol diversity can be beneficial in meeting new requirements, but the ability to innovate without public scrutiny risks point solutions that optimise for specific cases, but that can accidentally disrupt operations of/in different parts of the network. The social contract that maintains the stability of the Internet relies on accepting common interworking specifications, and on it being possible to detect violations. It is important to find new ways of maintaining that community trust as increased use of transport header encryption limits visibility into transport behaviour.

- Impact on Benchmarking and Understanding Feature Interactions: An appropriate vantage point for observation, coupled with timing information about traffic flows, provides a valuable tool for benchmarking network devices, endpoint stacks, functions, and/or configurations. This can also help with understanding complex feature interactions. An inability to observe transport header information can make it harder 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. New approaches might have to be developed.

- Impact on Research and Development: Hiding transport header 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 have 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 increased use of transport header encryption 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 Group (ICCRG) and research publications in reviewing new transport mechanisms and assessing the impact of their deployment).

Observable transport header information might be useful to various stake holders. Other sets of stake holders have incentives to limit what can be observed. This document does not make recommendations about what information ought to be exposed, to whom it ought to be observable, or how this will be achieved. There are also design choices about where observable fields are placed. For example, one location could be a part of the transport header outside of the encryption envelope, another alternative is to carry the information in a network-layer option or extension header. New transport protocol designs ought to explicitly identify any fields that are intended to be observed, consider if there are alternative ways of providing the information, and reflect on the implications of observable fields being used by network devices, and how this might impact user privacy and protocol evolution when these fields become ossified.

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." Providing explicit information can help avoid traffic being inappropriately classified, impacting application performance. An appropriate balance will emerge over time as real instances of this tension are analysed [RFC7258]. This balance between information exposed and information hidden ought to be carefully considered when specifying new transport protocols.

9. Security Considerations

This document is about design and deployment considerations for transport protocols. Issues relating to security are discussed throughout this 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 or to modify an existing transport. 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. Preventing observation of headers provides an 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 costs of ossifying common headers, versus the potential benefits of exposing specific information that could be observed along the network path to provide tools to manage new variants of protocols.

Header encryption can provide confidentiality of some or all of the transport 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. Reduces visibility into transport metadata can limit the ability to measure and characterise traffic. It can also provide privacy benefits in some cases.

Extending the transport payload security context to also include the transport protocol header protects both information with the same key. A privacy concern would arise if this key was shared with a third party, e.g., providing access to transport header information to debug a performance issue, would also result in exposing the transport payload data to the same third party. Such risks would be mitigated using a layered security design that provides one domain of protection and associated keys for the transport payload and encrypted transport headers; and a separate domain of protection and associated keys for any observable transport header fields.

Exposed transport headers are sometimes utilised as a part of the information to detect anomalies in network traffic. "While PM is an attack, other forms of monitoring that might fit the definition of PM can be beneficial and not part of any attack, e.g., network management functions monitor packets or flows and anti-spam mechanisms need to see mail message content." [RFC7258]. 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 normalisation 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 can also form 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 might 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 cannot be predicted (see Section 5.1 of [RFC8085]). A receiver could also require additional information to be used as a part of a validation 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 against on-path attacks. An additional processing cost can be incurred when decryption has 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 header 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.

The Security and Privacy Considerations in the Framework for Large-Scale Measurement of Broadband Performance (LMAP) [RFC7594] contain considerations for Active and Passive measurement techniques and supporting material on measurement context.

Addition of observable transport information to the path increases the information available to an observer and may, when this information can be linked to a node or user, reduce the privacy of the user. See the security considerations of [RFC8558].

10. IANA Considerations

XX RFC ED - PLEASE REMOVE THIS SECTION XXX

This memo includes no request to IANA.

11. 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, Thomas Fossati, Mohamed Boucadair, Martin Thomson, David Black, and other members of 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, and the EU Stand ICT Call 4. The opinions expressed and arguments employed reflect only the authors' view. The European Commission is not responsible for any use that might be made of that information.

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

12. 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-06 (work in progress), July 2019.

[I-D.ietf-quic-transport]
Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport", draft-ietf-quic-transport-22 (work in progress), July 2019.

[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]
Wood, C., Enghardt, T., Pauly, T., Perkins, C., and K. Rose, "A Survey of Transport Security Protocols", draft-ietf-taps-transport-security-08 (work in progress), August 2019.

[I-D.ietf-tls-dtls13]
Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", draft-ietf-tls-dtls13-38 (work in progress), May 2020.

[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.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.

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

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

[PAM-RTT] Trammell, B. and M. Kuehlewind, "Revisiting the Privacy Implications of Two-Way Internet Latency Data (in Proc. PAM 2018)", March 2018.

[Quic-Trace]
"https:QUIC trace utilities //github.com/google/quic-trace".

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[RFC2410] Glenn, R. and S. Kent, "The NULL Encryption Algorithm and Its Use With IPsec", RFC 2410, DOI 10.17487/RFC2410, November 1998, <https://www.rfc-editor.org/info/rfc2410>.

[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>.

[RFC3449] Balakrishnan, H., Padmanabhan, V., Fairhurst, G., and M. Sooriyabandara, "TCP Performance Implications of Network Path Asymmetry", BCP 69, RFC 3449, DOI 10.17487/RFC3449, December 2002, <https://www.rfc-editor.org/info/rfc3449>.

[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>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[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>.

[RFC5426] Okmianski, A., "Transmission of Syslog Messages over UDP", RFC 5426, DOI 10.17487/RFC5426, March 2009, <https://www.rfc-editor.org/info/rfc5426>.

[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>.

[RFC5795] Sandlund, K., Pelletier, G., and L-E. Jonsson, "The ROBust Header Compression (ROHC) Framework", RFC 5795, DOI 10.17487/RFC5795, March 2010, <https://www.rfc-editor.org/info/rfc5795>.

[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>.

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

[RFC7126] Gont, F., Atkinson, R., and C. Pignataro, "Recommendations on Filtering of IPv4 Packets Containing IPv4 Options", BCP 186, RFC 7126, DOI 10.17487/RFC7126, February 2014, <https://www.rfc-editor.org/info/rfc7126>.

[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>.

[RFC7413] Cheng, Y., Chu, J., Radhakrishnan, S., and A. Jain, "TCP Fast Open", RFC 7413, DOI 10.17487/RFC7413, December 2014, <https://www.rfc-editor.org/info/rfc7413>.

[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>.

[RFC7594] Eardley, P., Morton, A., Bagnulo, M., Burbridge, T., Aitken, P., and A. Akhter, "A Framework for Large-Scale Measurement of Broadband Performance (LMAP)", RFC 7594, DOI 10.17487/RFC7594, September 2015, <https://www.rfc-editor.org/info/rfc7594>.

[RFC7605] Touch, J., "Recommendations on Using Assigned Transport Port Numbers", BCP 165, RFC 7605, DOI 10.17487/RFC7605, August 2015, <https://www.rfc-editor.org/info/rfc7605>.

[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>.

[RFC7799] Morton, A., "Active and Passive Metrics and Methods (with Hybrid Types In-Between)", RFC 7799, DOI 10.17487/RFC7799, May 2016, <https://www.rfc-editor.org/info/rfc7799>.

[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>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[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>.

[RFC8462] Rooney, N. and S. Dawkins, Ed., "Report from the IAB Workshop on Managing Radio Networks in an Encrypted World (MaRNEW)", RFC 8462, DOI 10.17487/RFC8462, October 2018, <https://www.rfc-editor.org/info/rfc8462>.

[RFC8517] Dolson, D., Ed., Snellman, J., Boucadair, M., Ed., and C. Jacquenet, "An Inventory of Transport-Centric Functions Provided by Middleboxes: An Operator Perspective", RFC 8517, DOI 10.17487/RFC8517, February 2019, <https://www.rfc-editor.org/info/rfc8517>.

[RFC8546] Trammell, B. and M. Kuehlewind, "The Wire Image of a Network Protocol", RFC 8546, DOI 10.17487/RFC8546, April 2019, <https://www.rfc-editor.org/info/rfc8546>.

[RFC8548] Bittau, A., Giffin, D., Handley, M., Mazieres, D., Slack, Q., and E. Smith, "Cryptographic Protection of TCP Streams (tcpcrypt)", RFC 8548, DOI 10.17487/RFC8548, May 2019, <https://www.rfc-editor.org/info/rfc8548>.

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

[RFC8684] Ford, A., Raiciu, C., Handley, M., Bonaventure, O., and C. Paasch, "TCP Extensions for Multipath Operation with Multiple Addresses", RFC 8684, DOI 10.17487/RFC8684, March 2020, <https://www.rfc-editor.org/info/rfc8684>.

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

- Added some text on TLS story (additional input sought on relevant considerations).

- Section 2, paragraph 8 - changed to be clearer, in particular, added "Encryption with secure key distribution prevents"

- Flow label description rewritten based on PS/BCP RFCs.

- Clarify requirements from RFCs concerning the IPv6 flow label and highlight ways it can be used with encryption. (section 3.1.3)

- Add text on the explicit spin-bit work in the QUIC DT. Added greasing of spin-bit. (Section 6.1)

- Updated section 6 and added more explanation of impact on operators.

- Other comments addressed.

-05 Editorial pass and minor corrections noted on TSVWG list.

-06 Updated conclusions and minor corrections. Responded to request to add OAM discussion to Section 6.1.

-07 Addressed feedback from Ruediger and Thomas.

Section 2 deserved some work to make it easier to read and avoid repetition. This edit finally gets to this, and eliminates some duplication. This also moves some of the material from section 2 to reform a clearer conclusion. The scope remains focussed on the usage of transport headers and the implications of encryption - not on proposals for new techniques/specifications to be developed.

-08 Addressed feedback and completed editorial work, including updating the text referring to RFC7872, in preparation for a WGLC.

-09 Updated following WGLC. In particular, thanks to Joe Touch (specific comments and commentary on style and tone); Dimitri Tikonov (editorial); Christian Huitema (various); David Black (various). Amended privacy considerations based on SECDIR review. Emile Stephan (inputs on operations measurement); Various others.

Added summary text and refs to key sections. Note to editors: The section numbers are hard-linked.

-10 Updated following additional feedback from 1st WGLC. Comments from David Black; Tommy Pauly; Ian Swett; Mirja Kuehlewind; Peter Gutmann; Ekr; and many others via the TSVWG list. Some people thought that "needed" and "need" could represent requirements in the document, etc. this has been clarified.

-11 Updated following additional feedback from Martin Thomson, and corrections from other reviewers.

-12 Updated following additional feedback from reviewers.

-13 Updated following 2nd WGLC with comments from D.L.Black; T. Herbert; Ekr; and other reviewers.

-14 Update to resolve feedback to rev -13. This moves the general discussion of adding fields to transport packets to section 6, and discusses with reference to material in RFC8558.

-15 Feedback from D.L. Black, T. Herbert, J. Touch, S. Dawkins and M. Duke. Update to add reference to RFC7605. Clarify a focus on immutable transport fields, rather than modifying middleboxes with Tom H. Clarified Header Compression discusion only provides a list of examples of HC methods for transport. Clarified port usage with Tom H/Joe T. Removed some duplicated sentences, and minor edits. Added NULL-ESP. Improved after initial feedback from Martin Duke.

-16 Editorial comments from Mohamed Boucadair. Added DTLS 1.3.

Authors' Addresses

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

EMail: 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

EMail: csp@csperkins.org
URI: https://csperkins.org/