

Independent Submission
Request for Comments: 8517
Category: Informational
ISSN: 2070-1721

D. Dolson, Ed.
J. Snellman
M. Boucadair, Ed.
C. Jacquenet
Orange
February 2019

An Inventory of Transport-Centric Functions Provided by Middleboxes:
An Operator Perspective

Abstract

This document summarizes an operator's perception of the benefits that may be provided by intermediary devices that execute functions beyond normal IP forwarding. Such intermediary devices are often called "middleboxes".

RFC 3234 defines a taxonomy of middleboxes and issues in the Internet. Most of those middleboxes utilize or modify application-layer data. This document primarily focuses on devices that observe and act on information carried in the transport layer, and especially information carried in TCP packets.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc8517>.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Table of Contents

- 1. Introduction 3
 - 1.1. Operator Perspective 3
 - 1.2. Scope 4
- 2. Measurements 5
 - 2.1. Packet Loss 5
 - 2.2. Round-Trip Times 6
 - 2.3. Measuring Packet Reordering 6
 - 2.4. Throughput and Bottleneck Identification 7
 - 2.5. Congestion Responsiveness 7
 - 2.6. Attack Detection 8
 - 2.7. Packet Corruption 8
 - 2.8. Application-Layer Measurements 9
- 3. Functions beyond Measurement: A Few Examples 9
 - 3.1. NAT 9
 - 3.2. Firewall 10
 - 3.3. DDoS Scrubbing 10
 - 3.4. Implicit Identification 11
 - 3.5. Performance-Enhancing Proxies 11
 - 3.6. Network Coding 12
 - 3.7. Network-Assisted Bandwidth Aggregation 13
 - 3.8. Prioritization and Differentiated Services 13
 - 3.9. Measurement-Based Shaping 14
 - 3.10. Fairness to End-User Quota 14
- 4. IANA Considerations 14
- 5. Security Considerations 14
 - 5.1. Confidentiality and Privacy 14
 - 5.2. Active On-Path Attacks 15
 - 5.3. Improved Security 15
- 6. Informative References 16
- Acknowledgements 21
- Authors' Addresses 21

1. Introduction

From [RFC3234], "A middlebox is defined as any intermediary device performing functions other than the normal, standard functions of an IP router on the datagram path between a source host and destination host."

Middleboxes are usually (but not exclusively) deployed at locations permitting observation of bidirectional traffic flows. Such locations are typically points where leaf networks connect to the Internet, for example:

- o Where a residential or business customer connects to its service provider(s), which may include multihoming;
- o On the Gi interface where a Gateway General Packet Radio Service (GPRS) Support Node (GGSN) connects to a Packet Data Network (PDN) (Section 3.1 of [RFC6459]).

For the purposes of this document (and to be consistent with the definition in RFC 3234), middlebox functions may be found in routers and switches in addition to dedicated devices.

This document itemizes a variety of features provided by middleboxes and by ad hoc analysis performed by operators using packet analyzers.

Many of the techniques described in this document require stateful analysis of transport streams. A generic state machine is described in [PATH-STATE].

This document summarizes an operator's perception of the benefits that may be provided by middleboxes. A primary goal is to provide information to the Internet community to aid understanding of what might be gained or lost by decisions that may affect (or be affected by) middlebox operation in the design of new transport protocols. See Section 1.1 for more details.

1.1. Operator Perspective

Network operators are often the ones first called upon when applications fail to function properly, often with user reports about application behaviors (not about packet behaviors). Therefore, it isn't surprising that operators (wanting to be helpful) desire some visibility into flow information to identify how well the problem flows are progressing and how well other flows are progressing.

Advanced service functions (e.g., Network Address Translators (NATs), firewalls, etc.) [RFC7498] are widely used to achieve various objectives such as IP address sharing, firewalling, avoiding covert channels, detecting and protecting against ever-increasing Distributed Denial of Service (DDoS) attacks, etc. For example, environment-specific designs may require a number of service functions, such as those needed at the Gi interface of a mobile infrastructure [USE-CASE].

These sophisticated service functions are located in the network but also in service platforms or intermediate entities (e.g., Content Delivery Networks (CDNs)). Network maintenance and diagnostic operations are complicated, particularly when responsibility is shared among various players.

Network Providers are challenged to deliver differentiated services as a competitive business advantage while mastering the complexity of the applications, (continuously) evaluating the impacts on middleboxes, and enhancing customers' quality of experience.

Despite the complexity, removing all those service functions is not an option because they are used to address constraints that are often typical of the current (and changing) Internet. Operators must deal with constraints such as global IPv4 address depletion and support a plethora of services with different requirements for QoS, security, robustness, etc.

1.2. Scope

Although many middleboxes observe and manipulate application-layer content (e.g., session boarder controllers [RFC5853]), they are out of scope for this document, the aim being to describe middleboxes using transport-layer features. [RFC8404] describes the impact of pervasive encryption of application-layer data on network monitoring, protecting, and troubleshooting.

The current document is not intended to recommend (or prohibit) middlebox deployment. Many operators have found the value provided by middleboxes to outweigh the added cost and complexity; this document attempts to provide that perspective as a reference in discussion of protocol design trade-offs.

This document is not intended to discuss issues related to middleboxes. These issues are well known and are recorded in existing documents such as [RFC3234] and [RFC6269]. This document aims to elaborate on the motivations leading operators to enable such functions in spite of complications.

This document takes an operator perspective that measurement and management of transport connections is of benefit to both parties: the end user receives a better quality of experience, and the network operator improves resource usage, the former being a consequence of the latter.

This document does not discuss whether exposing some data to on-path devices for network-assistance purposes can be achieved by using in-band or out-of-band mechanisms.

2. Measurements

A number of measurements can be made by network devices that are either on-path or off-path. These measurements can be used either by automated systems or for manual network troubleshooting purposes (e.g., using packet-analysis tools). The automated systems can further be classified into two types: 1) monitoring systems that compute performance indicators for single connections or aggregates of connections and generate aggregated reports from them; and 2) active systems that make decisions also on how to handle packet flows based on these performance indicators.

Long-term trends in these measurements can aid an operator in capacity planning.

Short-term anomalies revealed by these measurements can identify network breakages, attacks in progress, or misbehaving devices/applications.

2.1. Packet Loss

It is very useful for an operator to be able to detect and isolate packet loss in a network.

Network problems and underprovisioning can be detected if packet loss is measurable. TCP packet loss can be detected by observing gaps in sequence numbers, retransmitted sequence numbers, and selective acknowledgement (SACK) options [RFC2018]. Packet loss can be detected per direction.

Gaps indicate loss upstream of the traffic observation point; retransmissions indicate loss downstream of the traffic observation point. SACKs can be used to detect either upstream or downstream packet loss (although some care needs to be taken to avoid misidentifying packet reordering as packet loss) and to distinguish between upstream versus downstream losses.

Packet-loss measurements on both sides of the measurement point are an important component in precisely diagnosing insufficiently dimensioned devices or links in networks. Additionally, packet losses are one of the two main ways for congestion to manifest, the others being queuing delay or Explicit Congestion Notification (ECN) [RFC3168]; therefore, packet loss is an important measurement for any middlebox that needs to make traffic handling decisions based on observed levels of congestion.

2.2. Round-Trip Times

The ability to measure partial-path round-trip times (RTTs) is valuable in diagnosing network issues (e.g., abnormal latency, abnormal packet loss). Knowing if latency is poor on one side of the observation point or the other provides more information than is available at either endpoint, which can only observe full RTTs.

For example, a TCP packet stream can be used to measure the RTT on each side of the measurement point. During the connection handshake, the SYN, SYN/ACK, and ACK timings can be used to establish a baseline RTT in each direction. Once the connection is established, the RTT between the server and the measurement point can only reliably be determined using TCP timestamps [RFC7323]. On the side between the measurement point and the client, the exact timing of data segments and ACKs can be used as an alternative. For this latter method to be accurate when packet loss is present, the connection must use selective acknowledgements.

In many networks, congestion will show up as increasing packet queuing, and congestion-induced packet loss will only happen in extreme cases. RTTs will also show up as a much smoother signal than the discrete packet-loss events. This makes RTTs a good way to identify individual subscribers for whom the network is a bottleneck at a given time or geographical sites (such as cellular towers) that are experiencing large-scale congestion.

The main limit of RTT measurement as a congestion signal is the difficulty of reliably distinguishing between the data segments being queued versus the ACKs being queued.

2.3. Measuring Packet Reordering

If a network is reordering packets of transport connections, caused perhaps by Equal-Cost Multipath (ECMP) misconfiguration (described in [RFC2991] and [RFC7690], for example), the endpoints may react as if packet loss is occurring and retransmit packets or reduce forwarding rates. Therefore, a network operator desires the ability to diagnose packet reordering.

For TCP, packet reordering can be detected by observing TCP sequence numbers per direction. See, for example, a number of standard packet-reordering metrics in [RFC4737] and informational metrics in [RFC5236].

2.4. Throughput and Bottleneck Identification

Although throughput to or from an IP address can be measured without transport-layer measurements, the transport layer provides clues about what the endpoints were attempting to do.

One way of quickly excluding the network as the bottleneck during troubleshooting is to check whether the speed is limited by the endpoints. For example, the connection speed might instead be limited by suboptimal TCP options, the sender's congestion window, the sender temporarily running out of data to send, the sender waiting for the receiver to send another request, or the receiver closing the receive window.

This data is also useful for middleboxes used to measure network quality of service. Connections, or portions of connections, that are limited by the endpoints do not provide an accurate measure of the network's speed and can be discounted or completely excluded in such analyses.

2.5. Congestion Responsiveness

Congestion control mechanisms continue to evolve. Tools exist that can interpret protocol sequence numbers (e.g., from TCP or RTP) to infer the congestion response of a flow. Such tools can be used by operators to help understand the impact of specific transport protocols on other traffic that shares their network. For example, packet sequence numbers can be analyzed to help understand whether an application flow backs off its load in the face of persistent congestion (as TCP does) and, hence, whether the behavior is appropriate for sharing limited network capacity.

These tools can also be used to determine whether mechanisms are needed in the network to prevent flows from acquiring excessive network capacity under severe congestion (e.g., by deploying rate limiters or network transport circuit breakers [RFC8084]).

2.6. Attack Detection

When an application or network resource is under attack, it is useful to identify this situation from the network perspective, upstream of the attacked resource.

Although detection methods tend to be proprietary, attack detection from within the network may comprise:

- o Identifying uncharacteristic traffic volumes or sources;
- o Identifying congestion, possibly using techniques in Sections 2.1 and 2.2;
- o Identifying incomplete connections or transactions, from attacks that attempt to exhaust server resources;
- o Fingerprinting based on whatever available fields are determined to be useful in discriminating an attack from desirable traffic.

Two trends in protocol design will make attack detection more difficult:

- o The desire to encrypt transport-layer fields;
- o The desire to avoid statistical fingerprinting by adding entropy in various forms.

While improving privacy, those approaches may hinder attack detection.

2.7. Packet Corruption

One notable source of packet loss is packet corruption. This corruption will generally not be detected until the checksums are validated by the endpoint and the packet is dropped. This means that detecting the exact location where packets are lost is not sufficient when troubleshooting networks. An operator would like to find out where packets are being corrupted. IP and TCP checksum verification allows a measurement device to correctly distinguish between upstream packet corruption and normal downstream packet loss.

Transport protocol designers should consider whether a middlebox will be able to detect corrupted or tampered packets.

2.8. Application-Layer Measurements

Information about network health may also be gleaned from application-layer diagnosis, such as:

- o DNS response times and retransmissions calculated by correlating answers to queries;
- o Various protocol-aware voice and video quality analyses.

Could this type of information be provided in a transport layer?

3. Functions beyond Measurement: A Few Examples

This section describes features provided by on-path devices that go beyond measurement by modifying, discarding, delaying, or prioritizing traffic.

3.1. NAT

Network Address Translators (NATs) allow multiple devices to share a public address by dividing the transport-layer port space among the devices.

NAT behavior recommendations are found for UDP in BCP 127 [RFC4787] and for TCP in BCP 142 [RFC7857].

To support NAT, there must be transport-layer port numbers that can be modified by the NAT. Note that required fields (e.g., port numbers) are visible in all IETF-defined transport protocols.

The application layer must not assume the port number was left unchanged (e.g., by including it in a checksum or signing it).

Address sharing is also used in the context of IPv6 transition. For example, DS-Lite Address Family Transition Router (AFTR) [RFC6333], NAT64 [RFC6146], or A+P [RFC7596][RFC7597] are features that are enabled in the network to allow for IPv4 service continuity over an IPv6 network.

Further, because of some multihoming considerations, IPv6 prefix translation may be enabled by some enterprises by means of IPv6-to-IPv6 Network Prefix Translation (NPTv6) [RFC6296].

3.2. Firewall

Firewalls are pervasive and essential components that inspect incoming and outgoing traffic. Firewalls are usually the cornerstone of a security policy that is enforced in end-user premises and other locations to provide strict guarantees about traffic that may be authorized to enter/leave the said premises, as well as end users who may be assigned different clearance levels regarding which networks and portions of the Internet they access.

An important aspect of a firewall policy is differentiating internally initiated from externally initiated communications.

For TCP, this is easily done by tracking the TCP state machine. Furthermore, the ending of a TCP connection is indicated by RST or FIN flags.

For UDP, the firewall can be opened if the first packet comes from an internal user, but the closing is generally done by an idle timer of arbitrary duration, which might not match the expectations of the application.

Simple IPv6 firewall capabilities for customer premises equipment (both stateless and stateful) are described in [RFC6092].

A firewall functions better when it can observe the protocol state machine, described generally by "Transport-Independent Path Layer State Management" [PATH-STATE].

3.3. DDoS Scrubbing

In the context of a DDoS attack, the purpose of a scrubber is to discard attack traffic while permitting useful traffic. Such a mitigator is described in [DOTS].

When attacks occur against constrained resources, some traffic will be discarded before reaching the intended destination. A user receives better experience and a server runs more efficiently if a scrubber can discard attack traffic, leaving room for legitimate traffic.

Scrubbing must be provided by an on-path network device, because neither endpoint of a legitimate connection has any control over the source of the attack traffic.

Source-spoofed DDoS attacks can be mitigated at the source using BCP 38 [RFC2827], but it is more difficult if source address filtering cannot be applied.

In contrast to devices in the core of the Internet, middleboxes statefully observing bidirectional transport connections can reject source-spoofed TCP traffic based on the inability to provide sensible acknowledgement numbers to complete the three-way handshake. Obviously, this requires middlebox visibility into a transport-layer state machine.

Middleboxes may also scrub on the basis of statistical classification: testing how likely a given packet is to be legitimate. As protocol designers add more entropy to headers and lengths, this test becomes less useful, and the best scrubbing strategy becomes random drop.

3.4. Implicit Identification

In order to enhance the end user's quality of experience, some operators deploy implicit identification features that rely upon the correlation of network-related information to access some local services. For example, service portals operated by some operators may be accessed immediately by end users without any explicit identification for the sake of improved service availability. This is doable thanks to on-path devices that inject appropriate metadata that can be used by the remote server to enforce per-subscriber policies. The information can be injected at the application layer or at the transport layer (when an address-sharing mechanism is in use).

An experimental implementation using a TCP option is described in [RFC7974].

For the intended use of implicit identification, it is more secure to have a trusted middlebox mark this traffic than to trust end-user devices.

3.5. Performance-Enhancing Proxies

Performance-Enhancing Proxies (PEPs) can improve performance in some types of networks by improving packet spacing or generating local acknowledgements; they are most commonly used in satellite and cellular networks. Transport-Layer PEPs are described in Section 2.1.1 of [RFC3135].

PEPs allow central deployment of congestion control algorithms more suited to the specific network, most commonly for delay-based congestion control. More advanced TCP PEPs deploy congestion control systems that treat all of a single end user's TCP connections as a single unit, improving fairness and allowing faster reaction to

changing network conditions. For example, it was reported that splitting the TCP connections in some network accesses can result in a halved page downloading time [ICCRG].

Local acknowledgements generated by PEPs speed up TCP slow start by splitting the effective latency, and they allow for retransmissions to be done from the PEP rather than from the actual sender. Local acknowledgements will also allow a PEP to maintain a local buffer of data appropriate to the actual network conditions, whereas the actual endpoints would often send too much or too little.

A PEP function requires transport-layer fields that allow chunks of data to be identified (e.g., TCP sequence numbers), acknowledgements to be identified (e.g., TCP ACK numbers), and acknowledgements to be created from the PEP.

Note that PEPs are only useful in some types of networks. In particular, PEPs are very useful in contexts wherein the congestion control strategies of the endpoints are inappropriate for the network connecting them. That being said, poor design can result in degraded performances when PEPs are deployed.

3.6. Network Coding

Network Coding is a technique for improving transmission performance over low-bandwidth, long-latency links such as some satellite links. Coding may involve lossless compression and/or adding redundancy to headers and payload. A Network Coding Taxonomy is provided by [RFC8406]; an example of end-to-end coding is FECFRAME [RFC6363]. It is typically deployed with network-coding gateways at each end of those links, with a network-coding tunnel between them via the slow/lossy/long-latency links.

Network-coding implementations may be specific to TCP, taking advantage of known properties of the protocol.

The network-coding gateways may employ some techniques of PEPs, such as creating acknowledgements of queued data, removing retransmissions, and pacing data rates to reduce queue oscillation.

The interest in more network coding in some specific networks is discussed in [SATELLITES].

Note: This is not to be confused with transcoding, which performs lossy compression on transmitted media streams and is not in scope for this document.

3.7. Network-Assisted Bandwidth Aggregation

The Hybrid Access Aggregation Point is a middlebox that allows customers to aggregate the bandwidth of multiple access technologies.

One of the approaches uses Multipath TCP (MPTCP) proxies [TCP-CONVERT] to forward traffic along multiple paths. The MPTCP proxy operates at the transport layer while being located in the operator's network.

The support of multipath transport capabilities by communicating hosts remains a privileged target design so that such hosts can directly use the available resources provided by a variety of access networks they can connect to. Nevertheless, network operators do not control end hosts, whereas the support of MPTCP by content servers remains marginal.

Network-assisted MPTCP deployment models are designed to facilitate the adoption of MPTCP for the establishment of multipath communications without making any assumption about the support of MPTCP capabilities by communicating peers. Network-assisted MPTCP deployment models rely upon MPTCP Conversion Points (MCPs) that act on behalf of hosts so that they can take advantage of establishing communications over multiple paths [TCP-CONVERT].

Note there are cases when end-to-end MPTCP cannot be used even though both client and server are MPTCP-compliant. An MPTCP proxy can provide multipath utilization in these cases. Examples of such cases are listed below:

1. The use of private IPv4 addresses in some access networks. Typically, additional subflows cannot be added to the MPTCP connection without the help of an MCP.
2. The assignment of IPv6 prefixes only by some networks. If the server is IPv4-only, IPv6 subflows cannot be added to an MPTCP connection established with that server, by definition.
3. Subscription to some service offerings is subject to volume quota.

3.8. Prioritization and Differentiated Services

Bulk traffic may be served with a higher latency than interactive traffic with no reduction in throughput. This fact allows a middlebox function to improve response times in interactive applications by prioritizing, policing, or remarking interactive transport connections differently from bulk-traffic transport

connections. For example, gaming traffic may be prioritized over email or software updates. Configuration guidelines for Diffserv service classes are discussed in [RFC4594].

Middleboxes may identify different classes of traffic by inspecting multiple layers of header and length of payload.

3.9. Measurement-Based Shaping

Basic traffic-shaping functionality requires no transport-layer information. All that is needed is a way of mapping each packet to a traffic shaper quota. For example, there may be a rate limit per 5-tuple or per subscriber IP address. However, such fixed traffic shaping rules are wasteful as they end up rate-limiting traffic even when the network has free resources available.

More advanced traffic-shaping devices use transport-layer metrics described in Section 2 to detect congestion on either a per-site or a per-user level and use different traffic-shaping rules when congestion is detected [RFC3272]. This type of device can overcome limitations of downstream devices that behave poorly (e.g., by excessive buffering or suboptimally dropping packets).

3.10. Fairness to End-User Quota

Several service offerings rely upon a volume-based charging model (e.g., volume-based data plans offered by cellular providers). Operators may assist end users in conserving their data quota by deploying on-path functions that shape traffic that would otherwise be aggressively transferred.

For example, a fast download of a video that won't be viewed completely by the subscriber may lead to quick exhaustion of the data quota. Limiting the video download rate conserves quota for the benefit of the end user. Also, discarding unsolicited incoming traffic prevents the user's quota from being unfairly exhausted.

4. IANA Considerations

This document has no IANA actions.

5. Security Considerations

5.1. Confidentiality and Privacy

This document intentionally excludes middleboxes that observe or manipulate application-layer data.

The measurements and functions described in this document can all be implemented without violating confidentiality [RFC6973]. However, there is always the question of whether the fields and packet properties used to achieve operational benefits may also be used for harm.

In particular, the question is what confidentiality is lost by exposing transport-layer fields beyond what can be learned by observing IP-layer fields:

- o Sequence numbers: an observer can learn how much data is transferred.
- o Start/Stop indicators: an observer can count transactions for some applications.
- o Device fingerprinting: an observer may be more easily able to identify a device type when different devices use different default field values or options.

5.2. Active On-Path Attacks

An on-path attacker being able to observe sequence numbers or session identifiers may make it easier to modify or terminate a transport connection. For example, observing TCP sequence numbers allows generation of a RST packet that terminates the connection. However, signing transport fields softens this attack. The attack and solution are described for the TCP authentication option [RFC5925]. Still, an on-path attacker can also drop the traffic flow.

5.3. Improved Security

Network maintainability and security can be improved by providing firewalls and DDoS mechanisms with some information about transport connections. In contrast, it would be very difficult to secure a network in which every packet appears unique and filled with random bits (from the perspective of an on-path device).

Some features providing the ability to mitigate/filter attacks owing to a network-assisted mechanism will therefore improve security -- e.g., by means of Distributed-Denial-of-Service Open Threat Signaling (DOTS) [DOTS-SIGNAL].

6. Informative References

- [DOTS] Mortensen, A., Andreasen, F., Reddy, T., Compton, R., and N. Teague, "Distributed-Denial-of-Service Open Threat Signaling (DOTS) Architecture", Work in Progress, draft-ietf-dots-architecture-07, September 2018.
- [DOTS-SIGNAL] Reddy, T., Boucadair, M., Patil, P., Mortensen, A., and N. Teague, "Distributed Denial-of-Service Open Threat Signaling (DOTS) Signal Channel Specification", Work in Progress, draft-ietf-dots-signal-channel-25, September 2018.
- [ICCRG] Kuhn, N., "MPTCP and BBR performance over Internet satellite paths", IETF 100, 2017, <<https://datatracker.ietf.org/meeting/100/materials/slides-100-iccr-mptcp-and-bbr-performance-over-satcom-links-00>>.
- [PATH-STATE] Kuehlewind, M., Trammell, B., and J. Hildebrand, "Transport-Independent Path Layer State Management", Work in Progress, draft-trammell-plus-statefulness-04, November 2017.
- [RFC2018] Mathis, M., Mahdavi, J., Floyd, S., and A. Romanow, "TCP Selective Acknowledgment Options", RFC 2018, DOI 10.17487/RFC2018, October 1996, <<https://www.rfc-editor.org/info/rfc2018>>.
- [RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, DOI 10.17487/RFC2827, May 2000, <<https://www.rfc-editor.org/info/rfc2827>>.
- [RFC2991] Thaler, D. and C. Hopps, "Multipath Issues in Unicast and Multicast Next-Hop Selection", RFC 2991, DOI 10.17487/RFC2991, November 2000, <<https://www.rfc-editor.org/info/rfc2991>>.
- [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>>.
- [RFC3272] Awduche, D., Chiu, A., Elwalid, A., Widjaja, I., and X. Xiao, "Overview and Principles of Internet Traffic Engineering", RFC 3272, DOI 10.17487/RFC3272, May 2002, <<https://www.rfc-editor.org/info/rfc3272>>.
- [RFC4594] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, DOI 10.17487/RFC4594, August 2006, <<https://www.rfc-editor.org/info/rfc4594>>.
- [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>>.
- [RFC4787] Audet, F., Ed. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, DOI 10.17487/RFC4787, January 2007, <<https://www.rfc-editor.org/info/rfc4787>>.
- [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>>.
- [RFC5853] Hautakorpi, J., Ed., Camarillo, G., Penfield, R., Hawrylyshen, A., and M. Bhatia, "Requirements from Session Initiation Protocol (SIP) Session Border Control (SBC) Deployments", RFC 5853, DOI 10.17487/RFC5853, April 2010, <<https://www.rfc-editor.org/info/rfc5853>>.
- [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>>.

- [RFC6092] Woodyatt, J., Ed., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, DOI 10.17487/RFC6092, January 2011, <<https://www.rfc-editor.org/info/rfc6092>>.
- [RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146, April 2011, <<https://www.rfc-editor.org/info/rfc6146>>.
- [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>>.
- [RFC6296] Wasserman, M. and F. Baker, "IPv6-to-IPv6 Network Prefix Translation", RFC 6296, DOI 10.17487/RFC6296, June 2011, <<https://www.rfc-editor.org/info/rfc6296>>.
- [RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, DOI 10.17487/RFC6333, August 2011, <<https://www.rfc-editor.org/info/rfc6333>>.
- [RFC6363] Watson, M., Begen, A., and V. Roca, "Forward Error Correction (FEC) Framework", RFC 6363, DOI 10.17487/RFC6363, October 2011, <<https://www.rfc-editor.org/info/rfc6363>>.
- [RFC6459] Korhonen, J., Ed., Soininen, J., Patil, B., Savolainen, T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 6459, DOI 10.17487/RFC6459, January 2012, <<https://www.rfc-editor.org/info/rfc6459>>.
- [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>>.
- [RFC7323] Borman, D., Braden, B., Jacobson, V., and R. Scheffenegger, Ed., "TCP Extensions for High Performance", RFC 7323, DOI 10.17487/RFC7323, September 2014, <<https://www.rfc-editor.org/info/rfc7323>>.

- [RFC7498] Quinn, P., Ed. and T. Nadeau, Ed., "Problem Statement for Service Function Chaining", RFC 7498, DOI 10.17487/RFC7498, April 2015, <<https://www.rfc-editor.org/info/rfc7498>>.
- [RFC7596] Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the Dual-Stack Lite Architecture", RFC 7596, DOI 10.17487/RFC7596, July 2015, <<https://www.rfc-editor.org/info/rfc7596>>.
- [RFC7597] Troan, O., Ed., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, Ed., "Mapping of Address and Port with Encapsulation (MAP-E)", RFC 7597, DOI 10.17487/RFC7597, July 2015, <<https://www.rfc-editor.org/info/rfc7597>>.
- [RFC7690] Byerly, M., Hite, M., and J. Jaeggli, "Close Encounters of the ICMP Type 2 Kind (Near Misses with ICMPv6 Packet Too Big (PTB))", RFC 7690, DOI 10.17487/RFC7690, January 2016, <<https://www.rfc-editor.org/info/rfc7690>>.
- [RFC7857] Penno, R., Perreault, S., Boucadair, M., Ed., Sivakumar, S., and K. Naito, "Updates to Network Address Translation (NAT) Behavioral Requirements", BCP 127, RFC 7857, DOI 10.17487/RFC7857, April 2016, <<https://www.rfc-editor.org/info/rfc7857>>.
- [RFC7974] Williams, B., Boucadair, M., and D. Wing, "An Experimental TCP Option for Host Identification", RFC 7974, DOI 10.17487/RFC7974, October 2016, <<https://www.rfc-editor.org/info/rfc7974>>.
- [RFC8084] Fairhurst, G., "Network Transport Circuit Breakers", BCP 208, RFC 8084, DOI 10.17487/RFC8084, March 2017, <<https://www.rfc-editor.org/info/rfc8084>>.
- [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>>.
- [RFC8406] Adamson, B., Adjih, C., Bilbao, J., Firoiu, V., Fitzek, F., Ghanem, S., Lochin, E., Masucci, A., Montpetit, M-J., Pedersen, M., Peralta, G., Roca, V., Ed., Saxena, P., and S. Sivakumar, "Taxonomy of Coding Techniques for Efficient Network Communications", RFC 8406, DOI 10.17487/RFC8406, June 2018, <<https://www.rfc-editor.org/info/rfc8406>>.

[SATELLITES]

Kuhn, N. and E. Lochin, "Network coding and satellites", Work in Progress, draft-irtf-nwcrp-network-coding-satellites-02, November 2018.

[TCP-CONVERT]

Bonaventure, O., Boucadair, M., Gundavelli, S., and S. Seo, "0-RTT TCP Convert Protocol", Work in Progress, draft-ietf-tcpm-converters-04, October 2018.

[USE-CASE]

Napper, J., Stiemerling, M., Lopez, D., and J. Uttaro, "Service Function Chaining Use Cases in Mobile Networks", Work in Progress, draft-ietf-sfc-use-case-mobility-08, May 2018.

Acknowledgements

The authors thank Brian Trammell, Brian Carpenter, Mirja Kuehlewind, Kathleen Moriarty, Gorry Fairhurst, Adrian Farrel, and Nicolas Kuhn for their review and suggestions.

Authors' Addresses

David Dolson (editor)

Email: ddolson@acm.org

Juho Snellman

Email: [jsnell@iki.fi](mailto:jnell@iki.fi)

Mohamed Boucadair (editor)

Orange

4 rue du Clos Courtel

Rennes 35000

France

Email: mohamed.boucadair@orange.com

Christian Jacquenet

Orange

4 rue du Clos Courtel

Rennes 35000

France

Email: christian.jacquenet@orange.com