PANRG
Internet-Draft
Intended status: Informational
Expires: 2 January 2026

J. van Bommel
F. Wirz
T. Zaeschke, Ed.
ETH Zurich

1 July 2025

Guidelines for QUIC Multipath over SCION
draft-zaeschke-scion-quic-multipath-00

Abstract

This document provides informational guidance for using the Multipath Extension for QUIC with the SCION networking technology.

SCION is an inter-domain routing protocol that supports path-aware multi-path networking. The multiple paths and their associated path information offered by SCION provide opportunities as well as challenges for combining QUIC-MP with SCION.

This document explores various aspects of this combination, such as algorithms for congestion control, RTT estimation, and general application scenarios. In addition, it provides techniques and guidance to maintain the security of QUIC-MP and SCION, and to leverage path-aware multi-path networking with QUIC-MP.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://netsec-ethz.github.io/scion-quic-multipath_I-D/draft-zaeschke-scion-quic-multipath.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-zaeschke-scion-quic-multipath/.

Discussion of this document takes place on the WG Working Group mailing list (mailto:panrg@irtf.org), which is archived at https://datatracker.ietf.org/rg/panrg. Subscribe at https://www.ietf.org/mailman/listinfo/panrg/.

Source for this draft and an issue tracker can be found at https://github.com/netsec-ethz/scion-quic-multipath_I-D.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

This Internet-Draft will expire on 2 January 2026.

Copyright Notice

Table of Contents

1. Introduction

The Multipath Extension for QUIC [QUIC-MP], is an extension for the QUIC protocol that enables simultaneous usage of multiple paths for a single QUIC connection.

SCION ([SCION-CP], [SCION-DP]) is an inter-domain routing protocol that offers explicit path selection between two endpoints, typically from a large selection of paths, where paths have detailed information on traversed autonomous systems (ASes), links, router interfaces, and other information.

Despite their complementary goals, QUIC-MP and PANs have evolved largely in isolation. QUIC-MP has been designed with traditional IP-based routing in mind, where path changes are typically inferred from endpoint address changes (i.e., 4-tuples), and where routing is opaque to the transport layer. In contrast, Path-Aware Networks (PANs), such as SCION, enable informed path selection based on performance, disjointness, policy, or security requirements. This combination of QUIC-MP and SCION allows for optimizations, for example, for congestion control, RTT estimation, failure recovery, performance, and security. However, the slightly different assumptions on endpoint addresses (4-tuple + path ID vs 4-tuple + AS code + path) and path lifecycles (path abandon vs expiry, etc.) can cause some pitfalls.

The purpose of this document is to explore how QUIC-MP and SCION can interoperate, how we can leverage the path awareness offered by SCION, and suggestions on how to overcome challenges.

This document lists notable points when using QUIC-MP over SCION (Section 3), looks at different usage scenarios (Section 4), gives implementations considerations (Section 5) for library developers of SCION and QUIC-MP, and discusses general security considerations (Section 7).

While we provide guidelines for these areas, we do not discuss concrete algorithms, APIs, QUIC-MP or SCION implementations, or QUIC-MP user applications; these are considered out of scope.

Some considerations are independent of multipathing and may be directly applicable for using [QUIC-TRANSPORT] over SCION.

2. Terminology and Conventions

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

2.1. Terminology

We assume that the reader is familiar with the terminology used in [QUIC-TRANSPORT] and [QUIC-MP]. We also draw on the terminology of [PATH-VOCABULARY] and of SCION ([SCION-CP] and [SCION-DP]). For ease of reference, we have included some definitions here, but refer the reader to the references above for complete specifications of the relevant terminology.

Autonomous System (AS): An autonomous system is a network under common administrative control. For example, the network of an Internet service provider or organization can constitute an AS.

Endpoint: An endpoint is the start or end of a path, as defined in [PATH-VOCABULARY].

Inter-AS Link: A direct link between two external interfaces of two ASes.

Intra-AS Link: A direct link between two internal interfaces of a single AS. A direct link may contain several internal hops.

Link: General term that refers to "inter-AS links" and "intra-AS links".

Network Address: In traditional IP networks, this refers to the tuple of address and port of an endpoint. In SCION, the network address consists of an _ISD-AS identifier_ and a _host address_, typically expressed as [ISD-AS, Host].

Network Path: The network path is the sequence of network elements between the two endpoints (e.g., ASes, interfaces, internal links). In traditional IP networks, the network path is typically opaque to the endpoints.

QUIC-MP Path: Consists of the network address at each endpoint and a Path ID (see [QUIC-MP]). The Path ID allows having multiple logical paths for the same set of network addresses.

Path Metadata: Path metadata is additional data that is available to endpoints when they request a selection of paths to a destination. Path metadata is authenticated by the owner of each link, but is otherwise not verified. This data describes properties of traversed ASes and links, such as their identity or MTU.

Metadata Extension: SCION offers additional path metadata via extensions. The metadata includes properties such as bandwidth and latency. The extension is widely supported but not further discussed here as it is not specified in [SCION-CP] or [SCION-DP].

3. Overview of QUIC Multipath in Path-Aware Networks

The Multipath Extension for QUIC (QUIC-MP) is primarily designed for traditional IP networks, where each path is identified by a 4-tuple of local and remote IP addresses and ports. Routing is considered opaque to endpoints, and path changes are inferred indirectly through changes in the 4-tuple. However, the path between two endpoints may change unpredictably due to routing dynamics, which is not captured by the 4-tuple.

In SCION, endpoints can discover and select explicit network paths, which are described at the level of ASes and inter-AS links, and have associated metadata, such as the MTU.

The different underlying assumptions of QUIC-MP and SCION result in some mismatches, for instance:

* *Endpoint Ambiguity*: PANs such as SCION use composite addresses (e.g., ISD-AS + Host), where the host can be an IP address from a private IP range. This breaks the assumption that IP addresses are unique in the current network.

* *Path Identity Mismatch*: In QUIC-MP, paths are distinguished by 4-tuples and Path IDs. In SCION, two distinct physical paths may share the same 4-tuple, rendering transport-layer path tracking incomplete.

* *Routing and Path Lifecycle*: Paths in SCION can expire, be revoked, or be reissued, even when 4-tuple information is unchanged. This can interfere with RTT estimation, congestion control, and path validation logic if not properly accounted for.

However, SCION also opens up opportunities for enhancing multipath transport:

* *Explicit path selection* enables endpoints to choose disjoint paths, i.e., paths that do not share links or segments, to improve fault tolerance against link failures, or to increase aggregate throughput.

* *Path metadata* allows endpoints to prioritize paths with more suitable properties. For instance, low-latency and low-hop-count paths can be prioritized for RTT measurements, avoiding wasting

resources on probing paths with poorer characteristics. Or, path overlap or disjointness can easily be determined and used for congestion control.

* *Path-awareness* allows congestion control and RTT estimators to reset only when the underlying network path has actually changed, something not reliably detectable in traditional IP networks, where network path changes may occur due to routing, despite a constant 4-tuple.

* The availability of *stable network paths* in PANs allows the transport layer to distinguish between actual path changes and transient network conditions, enabling more accurate RTT estimation and congestion control.

4. Multipath Use Cases and Categorization

This section categorizes common use cases for multipath transport and highlights how PANs enhance each scenario. Many of these use cases can be combined to meet complex application requirements.

4.1. Fault Tolerance and Availability

Multipath transport can improve robustness against network failures in several ways:

* An endpoint can immediately send data over a predetermined backup path if it suspects or obtains a network notification that a primary path is faulty.

* In SCION, the network may send error messages that explicitly specify the faulty link. This allows selecting a backup path that circumvents the faulty link.

* By sending redundant traffic over multiple paths, an application can maintain continuity even if one path becomes unavailable.

The use of backup paths or paths for redundant sending can be further improved by analyzing paths for overlap and selecting disjoint paths. This reduces the likelihood of multiple paths failing simultaneously.

4.2. High Throughput

An application may aim to maximize the available bandwidth by spreading traffic across multiple paths. To optimize this, an application may:

* Select multiple paths with maximum disjointness.

* Select multiple paths such that disjointness is limited to sub-paths that are expected to have high bandwidth available.

* When congestion is detected, switch over entirely, or shift parts of traffic to an underutilized disjoint path to preserve the throughput.

4.3. Low Latency

Latency-sensitive applications benefit from selecting the fastest available path at any moment. In PANs, endpoints may estimate latency from explicit metadata or infer it from probing. Because in PANs paths are stable and explicitly selectable, the transport layer can maintain multiple low-latency options in parallel, and either transmit in parallel, or switch traffic to a different path once the latency starts to fluctuate.

Latency may be determined by RTT estimation, see Section 5.5.

For deadline sensitive applications, an algorithm as described in [DMTP] may be useful.

Instead of probing many paths at once, an implementation should probe only the most promising paths (based on the metadata). Probing many paths should also be avoided to avoid overloading individual links, and it may effectively be limited (except traceroute) by the available path IDs and connection IDs, see Section 7.2 of [QUIC-MP].

4.4. Policy-Driven Routing and Routing Constraints

Some applications or deployments may wish to avoid routing traffic through certain ASes, e.g. to ensure path diversity or to enforce routing policies. SCION enables this by making path selection explicit and verifiable.

4.5. Anonymity and Traffic Obfuscation

Multipath transport can be used to reduce the observability of traffic by distributing it across multiple network paths. In PANs, endpoints can explicitly select disjoint paths to minimize the risk that a single AS observes the full traffic flow.

Randomizing path selection and packet scheduling can help obscure traffic patterns. However, traffic characteristics such as packet timing, flow duration, or volume may still be linkable across paths. Mechanisms such as probing should therefore be designed and used such that it avoids creating identifiable patterns.

4.6. Gateways and Proxies

There are gateways and proxies (including VPN) that translate SCION traffic to IP traffic and back. These are a special case because they are not used together with a QUIC(-MP) implementation, instead they are, and should be, oblivious to QUIC traffic.

TODO These, along with NATs, will be discussed in a future version of this document.

5. Technical Considerations

Using QUIC or QUIC-MP over a PAN, such as SCION, changes some of the underlying assumptions. This provides certain benefits, such as additional information and control over paths, but also some challenges.

5.1. Addressing

SCION uses composite addresses (AS + IP + port), where the IP address can be from a private IP range. This breaks the assumption that IP addresses are globally unique.

5.1.1. Key Implications

QUIC-MP relies on the 4-tuple changes to trigger path validation. However, with SCION, the 4-tuple does not uniquely identify an endpoint. Two endpoints with identical IP/port could be in different ASes. An attacker could use endpoints with identical 4-tuple to reroute traffic to a different machine without triggering path validation, see Section 7.1 and Section 5.11.

5.1.2. Recommendations

* To prevent attackers circumventing path validation, a QUIC-MP implementation MUST ensure to trigger path validation when the network address of the destination changes; this includes IP, port and AS number. This protects against several attacks, see Section 7.1 and especially Section 7.1.4.

There are several ways to achieve this, for example:

* Adapt the QUIC-MP library to be aware of the AS number in SCION network addresses.

* If the network address is available as a single "object", the SCION layer can extend this with the AS code and the QUIC-MP implementation must only ensure to compare the whole object instead of port and IP separately.

* The SCION implementation could detect cases where only the AS changes and then mangle the port or IP to trigger a path validation in the QUIC-MP layer. This may be a pragmatic solution but is discouraged because:

  - Managing paths in the SCION layer is not trivial because it requires synchronizing the lifecycle of SCION paths and QUIC-MP paths, e.g., knowing when is a path valid or when is it closed in QUIC-MP.

  - It creates opportunities for memory exhaustion attacks (for storing the mapping of mangled IP/port).

  - It reports a wrong IP/port to the application.

5.2. Interoperability of QUIC-MP Path ID and Network Paths

The identification of "paths" varies between QUIC, QUIC-MP and SCION.

* [QUIC-TRANSPORT] uses a 4-tuple of local/remote IP/port to distinguish paths.

* [QUIC-MP] extends the 4-tuple with a path ID to distinguish logical paths (connections).

* SCION can distinguish paths based on the physical inter-domain network path and additional properties, such as an expiration time (the latter may or may not be used to distinguish path instances).

A path change occurs when at least one of the router interfaces changes. The network address may stay the same.

With NAT rebinding, as described in Section 5.2 of [QUIC-MP], the path can change, but not without changing the SCION network address (IP, port, AS), so this case is not a concern.

Path change detection is required to trigger certain actions, such as resetting congestion control or RTT estimation algorithms. See also Section 5.4 and Section 5.5. When using a PAN such as SCION, it is important to trigger these actions even if the full network address (4-tuple + AS) stays the same.

Alternatively, the system can be implemented in a way so that uncontrolled path changes cannot occur. This is possible because path changes can only be initiated by endpoints. However, this has some limitation if one of the endpoints is not aware of transporting QUIC, for example a SCION gateway or proxy, see Section 4.6.

Implementations should try to maintain a 1:1 mapping between QUIC-MP path IDs and SCION network paths. However, this is not always possible or useful:

* A SCION network path may expire. Replacing a path with an identical new path (except for the expiration date), should be allowed without triggering algorithm reset. Alternatively, refresh can be handled by the path selector, see Section 5.9.

* A SCION implementation, when used with QUIC-MP should be configured such that every SCION network path is used for exactly one QUIC-MP path ID. However, it may not always be possible or feasible to configure SCION implementations in this way, for example. when they are part of a SCION gateway or proxy, and are unaware of transporting QUIC, see Section 4.6.

* A SCION path should be allowed to be reused, e.g., they may be assigned to one QUIC path ID, and when that path ID is closed, the path must be allowed to be assigned to another path ID. This should not cause any problems except for the marginal complexity of managing the associated state with a path ID.

5.2.1. Key Implications

If a path change occurs undetected, the QUIC-MP layer cannot reset congestion control (Section 5.4) or RTT estimation (Section 5.5). This is undesirable but not worse than traditional IP based non-PAN transport where routes can change without the endpoints learning about it.

5.2.2. Recommendations

* Congestion control and RTT estimation algorithms should be designed to gracefully handle path changes that don’t trigger a reset, unless it can be guaranteed that both SCION endpoints are configured to prevent automatic path changes.

* Within a QUIC-MP session, every SCION network path should be used only with one path ID. However, it may be reused if the path was abandoned or closed.

* Changes of the network path (while the network address stays the same), except for expired paths being renewed, should trigger algorithm reset (CC, RTT estimate), see Section 5.1 of [QUIC-MP].

Analogous to Section 5.1, except for replacing "AS" with "network path". We list them here again because the implications of not following the recommendation are much weaker and may be considered acceptable. Recommendations:

* Adapt the QUIC-MP library to be aware of the full network path, including router interfaces.

* If the network address is available as a single "object", the SCION layer can extend this with the network path (possibly excluding the expiration date), and the QUIC-MP implementation must only ensure to compare the whole object instead of port and IP separately.

* The SCION implementation could detect cases where only the router interfaces change and then mangle the port or IP to trigger a path validation in the QUIC-MP layer. This may be a pragmatic solution but is discouraged, because:

  - Managing paths in the SCION layer is not trivial because it requires synchronizing the lifecycle of SCION paths and QUIC-MP paths, e.g., knowing when is a path valid or when is it closed in QUIC-MP.

  - It creates opportunities for memory exhaustion attacks (for storing the mapping of mangled IP/port).

  - It reports a wrong IP/port to the application.

5.3. Initial Handshake

[QUIC-TRANSPORT] requires that there is no connection migration during the initial handshake, and that there are no other packets sent (including probing packets) during the initial handshake, see Section 9 of [QUIC-TRANSPORT], paragraphs 2 and 3.

5.3.1. Key Implications

Changing the path during handshake would violate Section 9 of [QUIC-TRANSPORT]:

> The design of QUIC relies on endpoints retaining a stable address for the duration of the handshake. An endpoint MUST NOT initiate connection migration before the handshake is confirmed, as defined in Section 4.1.2 of [QUIC-TLS].

5.3.2. Recommendations

* A SCION implementation should not automatically change network paths switch without explicit request by the QUIC(-MP) layer. The only exception allowed is replacing an expiring path with an new path that is identical except for the expiration time. We also need to ensure this for gateways etc, see Section 4.6.

5.4. Congestion Control

Following Section 5.1 of [QUIC-MP], the CC algorithm should be reset when the 4-tuple of a QUIC path changes. With SCION, 4-tuples are not sufficient to identify paths, see Section 5.2.

To avoid missing a path change, the SCION implementation should never change a network path unless explicitly instructed by the QUIC-MP implementation, see Section 5.1.2.

5.4.1. Coupled Congestion Control

Section 5.3 of [QUIC-MP] mentions coupled congestion control algorithms, such as [CC-MULTIPATH-TCP]. [CC-MULTIPATH-TCP] states:

> "One of the prominent problems is that running existing algorithms such as standard TCP independently on each path would give the multipath flow more than its fair share at a bottleneck link traversed by more than one of its subflows.".

This can be avoided in PANs, such as SCION, through link-level analysis of paths and selecting paths that do not share a bottleneck link. Instead, this bottleneck knowledge can be used to effectively use separate congestion control for each path. Alternatively, a CC algorithm could be employed that focuses on known shared links (which may be bottlenecks).

There are several congestion control algorithms proposed in literature, e.g., LIA, OLIA, BALIA and RSF. These combine congestion control with path selection algorithms. For simplicity, we suggest separating concerns in terms of congestion control and path selection. This allows us to better tailor the solutions to the different usage scenarios.

The proposition is to use non-coupled congestion control per path, tailored for each use case in Section 4, and use separate independent path selection algorithms.

CC algorithms can also benefit from the SCION metadata extension that provides bandwidth and latency data for each link on a network path.

5.4.2. Key Implications

A network path change goes unnoticed in case a SCION implementation changes a path that happens to have the same IP/port for both endpoints.

Congestion control (CC) algorithms can also benefit from exact knowledge of a path:

* When using multiple paths, a CC algorithm can access path metadata as to if and where the paths overlap and some of the properties of the overlapping sections.

* CC algorithms should be notified of every path change, allowing them to reset only when necessary. A reset may not be necessary if the network path remains the same and only the IP or port of an endpoint changes. This can make sense if any congestion is assumed to be on the network path rather than behind the remote IP/port (e.g., behind a proxy).

See also Section 5.3 of [QUIC-MP].

5.4.3. Recommendations

* Congestion control algorithms should be reset when the network path changes (beyond 4-tuple). This is best achieved by ensuring that the network path changes only in conjunction with QUIC path migration events.

Congestion control algorithms can also benefit from exact knowledge of a network path:

* Congestion control algorithms should use the path metadata to detect and, if necessary and possible, avoid overlap with other paths. Congestion control can then be simplified to work independently for each path.

* Path selection algorithms should try to avoid multiple path that share bottleneck links.

5.5. RTT Estimation

Similarly to congestion control, and following Section 5.1 of [QUIC-MP], the RTT estimation algorithm should be reset when the 4-tuple of a QUIC path changes. As described in Section 5.4 this can be avoided by forbidding SCION implementations to change a network path unless instructed otherwise by the QUIC-MP implementation.

5.5.1. Key Implications

If a path change occurs undetected, the QUIC-MP layer may fail to reset RTT estimation. This is undesirable but not worse than traditional IP based non-PAN transport where routes can change without the endpoints ever learning about it.

5.5.2. Recommendations

* RTT-algorithms should be reset when the network path changes (beyond 4-tuple). This is best achieved by ensuring that the network path changes only when requested by QUIC.

Round-trip time estimation algorithms can also benefit from exact knowledge of a path:

* An implementation may use SCION’s SCMP traceroute (Section 6 of [SCION-CP]) to measure the latency of individual links and then use this information to select new network paths that favor low latency links and avoid high latency links. See also Section 5.9.2.

* An implementation could use the SCION metadata extension to get propagation latency information of links in a path without having to measure it. This latency does not include queing latency but may in many cases be sufficient for practical use.

5.6. MTU Discovery

The MTU may be used to calculate the available payload size. SCION inserts an additional header (Section 2 of [SCION-DP]) into each packet. The header size depends on the IP family (e.g., IPv4 vs IPv6 addresses) and on the "length" of the path, i.e., the number of ASes that are traversed. This must be taken into account when calculating the available payload size.

The difference between typical MTU (1500 bytes) and QUIC’s required packet size (1200 bytes) is sufficient for typical real-world SCION headers.

PMTU discovery Section 14.3 of [QUIC-TRANSPORT] can be used to discover or verify MTU sizes. However, path metadata MTU can (at least on the client side) be used to preselect paths with desirable MTU values.

In SCION, when an endpoint requests a network path, it will be provided with MTU information for every hop on a path, see also Section 5.6 and Section 4.4 of [SCION-CP]. However, in SCION, paths are typically only requested by client endpoints, not by server endpoints.

There are several ways for a server to determine the MTU. If a server wants to know the MTU, it may:

* Try to determine the MTU from the size of incoming packets.

* Use an algorithm to determine the MTU, see Path MTU Discovery in Section 14.3 of [QUIC-TRANSPORT] and Section 5.8 of [QUIC-MP].

* Try to look up the path to the client endpoint that is identical to the incoming path. However, this requires time and effort on the server side, and there is no guarantee that the incoming path is available in the local AS.

Also, note that the MTU information is authenticated but not verified, it may be incorrect due to misconfiguration or malicious ASes.

5.6.1. Key Implications and Recommendations

PMTU discovery for multi-path may be improved by using path metadata. PMTU will be explored more in detail in a future version of this document (*TODO*)).

5.7. Retransmission & PTO

See Section 5.6 of [QUIC-MP] and Section 5.7 of [QUIC-MP]. For retransmission, a SCION client or server may compare available paths and choose one or more paths that have minimum overlap with the current (unreliable) path.

5.8. Paths Having Different PMTU Sizes

Section 5.8 of [QUIC-MP] suggests determining a single MTU size in order to simplify retransmission.

5.8.1. Key Implications

Section 5.8 of [QUIC-MP] explains that the benefit of using a single MTU size is to simplify retransmission processing, as the content of lost packets initially sent on one path can be sent on another path without further frame scheduling adaptations.

5.8.2. Recommendations

* On the client, this can be facilitated by computing a viable minimum MTU size from all available network paths. However, it must be considered that these values are not verified.

* On the server, MTU values from path metadata are not available. The server may request these from the local AS, but the exact path may not be available (in SCION, different ASes may offer different sets of paths to their customers). Also, except for initially proposing a preferred address (Section 9.6 of [QUIC-TRANSPORT]), new paths must be opened by the client, not the server, see Section 9 of [QUIC-TRANSPORT].

5.9. Path Selection

The path selection component is responsible for requesting paths to a destination, ordering the path based on policy and preferences, using them when new QUIC-paths are opened, and retiring them or listing them for reuse when they are closed.

5.9.1. Dynamic Approach

A dynamic approach could start with using low latency paths. If the connection appears to be long lasting, it could start (and keep) adding additional paths as long as the traffic increases.

As an example, if the algorithm detects traffic that lasts for at least one second and transfers at least 100MB of traffic, the algorithm could trigger creation of additional QUIC paths.

5.9.2. Bottleneck Detection

If live traffic information is not available, bottleneck detection can help to identify links that should be avoided. In PANs, this can be achieved using approaches such as [UMCC].

An alternative is to use SCION's SCMP traceroute command (Section 6 of [SCION-CP]) to measure the latency between two consecutive AS border routers. The measured latency can be compared to earlier measurements or to the latency given in the path metadata. Discrepancies can be an indication of high traffic volume and queueing on the measured link.

While traceroute may be useful, it should be used with care:

* traceroute traffic is not congestion controlled.

* It is clearly distinguishable from QUIC traffic, so it may affect anonymity.

5.9.3. Key Implications

Path selection is a key feature of SCION and PANs in general. For more details, see [SCION-CP] and [SCION-DP].

5.9.4. Recommendations

In order to manage paths effectively, the path selection algorithm probably requires access to the following fields and events:

* initial_max_path_id (Section 2.1 of [QUIC-MP])

* MAX_PATH_ID frames (Section 4.6 of [QUIC-MP]

* PATH_AVAILABLE and PATH_BACKUP, see Section 3.3 of [QUIC-MP],

* PATH_ABANDON Section 3.4 of [QUIC-MP].

Moreover, path selection must exclude paths whose MTU is too small to guarantee 1200 bytes MTU payload for QUIC packets. The effective MTU also depends on the length of the paths.

5.10. Packet Scheduling

Packet scheduling helps to distribute the transfer load efficiently over multiple paths, see also Section 5.5 of [QUIC-MP].

5.10.1. Key Implications

SCION paths are stable, at least on the inter-AS level, i.e., they cannot change without initiative from the endpoints.

5.10.2. Recommendations

Path stability may simplify packet scheduling algorithms because the performance of individual QUIC-paths is more reliable if they cannot unexpectedly be rerouted.

5.11. Address Validation Token

Section 9.3 of [QUIC-TRANSPORT] specifies that a server is expected to send a new address validation token to a client following the successful validation of a new client address.

Potential challenges:

* If we adapt an implementation to use the full network address + path for identity, and if we use this to generate tokens, then we may end up generating many more or longer tokens.

* Clients may not know their IP address (e.g., NAT) and their IP address may change.

See discussion in https://github.com/quicwg/multipath/issues/550

See also Section 21.3 of [QUIC-TRANSPORT].

TODO This section will be completed in a future version of this document.

6. Summary of Recommendations

This memo is informational. However, we use [RFC2119] imperative language here for recommendations that are relevant to security or performance.

6.1. Recommendations for QUIC-MP Implementations

* To prevent attackers circumventing path validation, a QUIC-MP implementation MUST ensure to trigger path validation when the network address of the destination changes; this includes IP, port and AS number. This protects against several attacks, see Section 7.1 and especially Section 7.1.4.

  There are several ways to achieve this, for example:

  - Adapt the QUIC-MP library to be aware of the AS number in SCION network addresses.

- If the network address is available as a single "object", the SCION layer can extend this with the AS code, and the QUIC-MP implementation must only ensure to compare the whole object instead of port and IP separately.

- The SCION implementation could detect cases where only the AS changes and then mangle the port or IP to trigger a path validation in the QUIC-MP layer. This may be a pragmatic solution but is discouraged, because:

  o Managing paths in the SCION layer is not trivial because it requires synchronizing the lifecycle of SCION paths and QUIC-MP paths, e.g., knowing when is a path valid or when is it closed in QUIC-MP.

  o It creates opportunities for memory exhaustion attacks (for storing the mapping of mangled IP/port).

  o It reports a wrong IP/port to the application.

* A QUIC-MP implementations SHOULD be able to recognize network path changes beyond 4-tuple or AS changes. This enables resetting congestion control and RTT algorithms.

6.2. Recommendations for SCION Implementations

* A SCION implementation SHOULD NOT store or cache paths, especially not on the server side. This prevents memory exhaustion attacks, see {attack-memory-exhaustion}. This also avoids the problem of path lifecycle maintenance, i.e., determining which paths are still alive and which have been closed or abandoned. Sometimes, storing paths is inevitable, see Section 4.6. For security concerns, see also Section 7.1.

* When used with QUIC-MP, a SCION implementation MUST not change the network paths, possibly with the exception of refreshing expired paths. When a path stops working, the implementation should instead report an error to the QUIC(-MP) layer or time out silently.

6.3. Recommendations for both QUIC-MP and SCION Implementations

* A server should return packets on the same path on which they were received.

- Generally, a server SHOULD respond on the same path on which the data was originally requested, unless the new path has been validated. This ensures that the path does not violate the path policy of the client.

- A server SHOULD NOT create new paths. This is also stated in Section 9 of [QUIC-TRANSPORT]. Servers may communicate a preferred address after the initial handshake. However, it is recommended to avoid that because any new path may violate a client’s path policy.

- Returning probing packets on the same network path on which they were received: This greatly simplifies RTT estimation, see Section 5.5.

* Clients may replace expired or soon-to-expire paths with identical paths without performing path migration / validation.

* Within a QUIC-MP session, every SCION network path should be used only with one path ID. However, it may be reused if the path was abandoned or closed by QUIC. This is the responsibility of the path selection algorithm, regardless of whether it is considered part of SCION or part of QUIC-MP.

7. Security Considerations

The aim is that QUIC-MP over SCION retains all security properties of QUIC-MP and SCION. However, this requires some implementation changes and additional consideration regarding:

* endpoint identity: a 4-tuple is not sufficient to identify an endhost;

* network path authenticity: paths may be forged by malicious clients;

* path probing patterns may expose user intentions or identity.

7.1. Path Injection

There are several potential attacks that build on injecting valid or invalid paths into the server-side software stack.

In summary, these attacks can be prevented by the recommendations listed in Section 6, specifically we recommend the following where possible:

1. SCION layers should avoid storing/caching paths and network addresses (beyond IP/port) internally. Instead, they should be given to the QUIC(-MP) layer or the application layer. That means that path information would only be accepted and retained if the QUIC(-MP) or application layer decides to do so.

2. SCION layers and QUIC(-MP) layers should interface by using network addresses that include all information that identifies an endpoint, including, for example, AS code. Any change in a network address (including the AS code) should trigger path validation.

Alternatives:

1. If paths and network addresses must be stored in the SCION layer, an alternative solution is to implement a form of signalling which would indicate that a packet is (or would be) rejected/ dropped by the QUIC(-MP) layer. These addresses and path from such packets should not be stored. However, to avoid connection drop, they should not be removed if they were previously used with a valid connection.

Examples of attacks include memory exhaustion attacks, traffic redirection attacks, and traffic amplification attacks.

7.1.1. Memory Exhaustion

An attacker may flood a server with packets that each have a different source network address. If these are stored in the SCION layer, they may cause memory exhaustion.

Mitigation: do not store state in the SCION layer, or implement a way to clean up state without affecting a valid connection.

7.1.2. Traffic Redirection to Different AS

An attacker may craft a packet that appears to originate from the same IP/port, but is located in a different AS than an existing connection. If the server’s SCION layer stores paths internally, and uses IP/port as key to look them up, then the new paths may replace the existing one, and outgoing traffic is redirected to the new paths and destination.

Mitigation:

* The QUIC(-MP) layer MUST trigger path validation if the network address changes, and must consider every attribute of the address, not just IP and port.

* If a packet is rejected by the QUIC(-MP) layer, the SCION layer MUST NOT add it to any local state (including not replacing existing state). This can be achieved trivially by not having state in the SCION layer.

7.1.3. Traffic Redirection over Different Path

An attacker may craft a path with a network address that is identical to an existing valid endpoint, but with a different path.

The new route may be invalid (e.g., contain nonexistent links) or faulty (contain links that are broken or have high latency or drop rate).

The new route may also work fine, but violate the client's path policy or be used for traffic analysis.

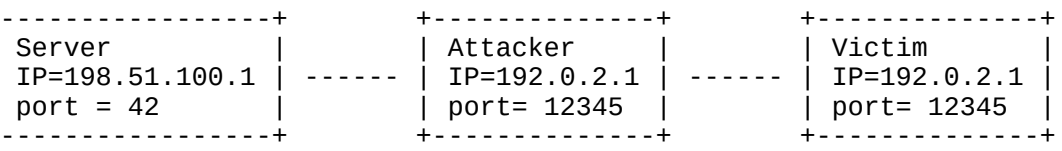

Figure 1: Example of non-unique IPs

This attack requires either spoofing of the client's IP address (when the attacker is in the same AS as the client) or injection of a path (which requires control over an AS that is en-route between the client and server).

Mitigation:

This is mitigated by the recommendation that path validation should always be triggered when the network address or path changes, even if the 4-tuple stays the same.

7.1.4. Traffic Amplification

An attacker may establish a connection with a server, request a large amount of data, and then inject a path that redirects to a victim that has the same IP/port, but in a different AS.

If the server-side QUIC-MP does not trigger path validation (because IP/port are the same), then it may implicitly accept the new path and send the requested data to a victim.

This attack requires the attacker to have control over an AS that is en-route between client (victim) and server.

Mitigation:

* A QUIC(-MP) library must consider all attributes (not just the 4-tuple) when checking for a change in the network address. This would then trigger path validation, and the attack can be averted.

* If a QUIC(-MP) library cannot compare additional attributes (e.g., legacy library), the SCION layer (server side) should have an option to perform port mangling or IP mangling: when the SCION layer detects a new network address that differs only in the AS number from a previously seen address (IP/port are the same), then it should perform IP/port mangling, i.e., reporting a modified IP or port to the QUIC(-MP) layer. This new IP/port would trigger a path validation or algorithm reset where required.

Caveats:

* Offering a mangled IP/port to the application may have implications for application correctness, such as displaying an unexpected IP/port.

Figure 2: Example of traffic amplification attack

7.2. Number of Open Paths

The number of open paths should be limited, see Section 7.2 of [QUIC-MP]. This is important in the context of applications that may open many paths in parallel.

Mitigation:

* Same as Section 7.2 of [QUIC-MP]: endpoints need to limit the maximum number of paths and might consider additional measures to limit the number of concurrent path validation processes, e.g., by pacing them out or limiting the number of path initiation attempts over a certain time period.

7.3. Probe Fingerprinting

An endpoint may probe multiple paths to determine the best path(s) for a given use case. One example of probing packets is packets that measure round-trip time (RTT).

Probing packets may be detected if they are sent in bulk, to the same destination, in regular intervals, and all with slightly different paths attached.

This can be used to fingerprint an endpoint or their intentions (applications may have unique intervals defined).

This can be mitigated by varying and generally reducing the number of probing packets, and by sending probing packets not en-block but time-shifted.

7.4. Additional Points

TODO: Complete this section in a future version of this document.

* Use multipathing for anonymity, see Section 4.

* See other attacks in Section 7.2.4 of [SCION-CP]?

8. IANA Considerations

This document has no IANA actions.

9. References

9.1. Normative References

[CC-ALGORITHMS]
Duke, M., Ed. and G. Fairhurst, Ed., "Specifying New Congestion Control Algorithms", BCP 133, RFC 9743, DOI 10.17487/RFC9743, March 2025, <https://www.rfc-editor.org/rfc/rfc9743>.

[CC-PRINCIPLES]
Floyd, S., "Congestion Control Principles", BCP 41, RFC 2914, DOI 10.17487/RFC2914, September 2000, <https://www.rfc-editor.org/rfc/rfc2914>.

[DCCP-UDPENCAP]
Phelan, T., Fairhurst, G., and C. Perkins, "DCCP-UDP: A Datagram Congestion Control Protocol UDP Encapsulation for NAT Traversal", RFC 6773, DOI 10.17487/RFC6773, November 2012, <https://www.rfc-editor.org/rfc/rfc6773>.

[MPTCP-ARCHITECTURE]
Ford, A., Raiciu, C., Handley, M., Barre, S., and J. Iyengar, "Architectural Guidelines for Multipath TCP Development", RFC 6182, DOI 10.17487/RFC6182, March 2011, <https://www.rfc-editor.org/rfc/rfc6182>.

[MTU-DISCOVERY]
Fairhurst, G., Jones, T., T端xen, M., R端ngeler, I., and T. V旦lker, "Packetization Layer Path MTU Discovery for Datagram Transports", RFC 8899, DOI 10.17487/RFC8899, September 2020, <https://www.rfc-editor.org/rfc/rfc8899>.

[QUIC-RECOVERY]
Iyengar, J., Ed. and I. Swett, Ed., "QUIC Loss Detection and Congestion Control", RFC 9002, DOI 10.17487/RFC9002, May 2021, <https://www.rfc-editor.org/rfc/rfc9002>.

[QUIC-TLS] Thomson, M., Ed. and S. Turner, Ed., "Using TLS to Secure QUIC", RFC 9001, DOI 10.17487/RFC9001, May 2021, <https://www.rfc-editor.org/rfc/rfc9001>.

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

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[UDP-GUIDELINES]
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/rfc/rfc8085>.

9.2. Informative References

[CC-MULTIPATH-TCP]
Raiciu, C., Handley, M., and D. Wischik, "Coupled Congestion Control for Multipath Transport Protocols", RFC 6356, DOI 10.17487/RFC6356, October 2011, <https://www.rfc-editor.org/rfc/rfc6356>.

[DMTP] John, T. and T. Riechard, "Deadline Aware Streams in QUIC Multipath", Work in Progress, Internet-Draft, draft-tjohn-quic-multipath-dmtp-01, 5 June 2025, <https://datatracker.ietf.org/doc/html/draft-tjohn-quic-multipath-dmtp-01>.

[OLIA] Khalili, R., Gast, N., Popovic, M., Upadhyay, U., and J. Le Boudec, "MPTCP is not pareto-optimal: performance issues and a possible solution", Proceedings of the 8th international conference on Emerging networking experiments and technologies, ACM , 2012.

[PATH-VOCABULARY]
Enghardt, R. and C. Kr辰henb端hl, "A Vocabulary of Path Properties", RFC 9473, DOI 10.17487/RFC9473, September 2023, <https://www.rfc-editor.org/rfc/rfc9473>.

[QUIC-ACKFREQUENCY]
Iyengar, J., Swett, I., and M. K端hlewind, "QUIC Acknowledgment Frequency", Work in Progress, Internet-Draft, draft-ietf-quic-ack-frequency-11, 28 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-quic-ack-frequency-11>.

[QUIC-MP] Liu, Y., Ma, Y., De Coninck, Q., Bonaventure, O., Huitema, C., and M. K端hlewind, "Multipath Extension for QUIC", Work in Progress, Internet-Draft, draft-ietf-quic-multipath-14, 23 April 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-quic-multipath-14>.

[SCION-CP] de Kater, C., Rustignoli, N., and S. Hitz, "SCION Control Plane", Work in Progress, Internet-Draft, draft-dekater-scion-controlplane-08, 3 March 2025, <https://datatracker.ietf.org/doc/html/draft-dekater-scion-controlplane-08>.

[SCION-DP] de Kater, C., Rustignoli, N., Hugly, J., and S. Hitz, "SCION Data Plane", Work in Progress, Internet-Draft, draft-dekater-scion-dataplane-05, 3 March 2025, <https://datatracker.ietf.org/doc/html/draft-dekater-scion-dataplane-05>.

[UMCC] Gartner, M. and D. Hausheer, "UMCC: Uncoupling Multipath Congestion Control through Shared Bottleneck Detection in Path-Aware Networks", Proceedings of the IEEE 49th Conference on Local Computer Networks (LCN) , 2024.

Acknowledgments

Thanks to the Path Aware Networking Research Group for discussion and feedback. Specifically, we would like to thank Kevin Meynell and Nicola Rustignoli from the Scion Association for their valuable input on several iterations of this document.

Authors' Addresses

Jelte van Bommel
ETH Zurich
Email: jelte.vanbommel@inf.ethz.ch

Francois Wirz
ETH Zurich
Email: wirzf@inf.ethz.ch

Tilmann Zaeschke (editor)
ETH Zurich
Email: tilmann.zaeschke@inf.ethz.ch