Multipath QUIC: Design and Evaluation

Quentin De Coninck, Olivier Bonaventure
quentin.deconinck@uclouvain.be

multipath-quic.org
Outline

• The QUIC protocol
• Designing Multipath for QUIC
• Experimental Design Evaluation
• Ongoing Work and Conclusion
QUIC: watisda?
QUIC = Quick UDP Internet Connection

- **TCP/TLS1.3 atop UDP**
  - >7% of the Internet traffic (YouTube, Chrome,...)
- **Stream multiplexing → HTTP/2 use case**
- **0-RTT establishment (most of the time)**
### QUIC Packet

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID</th>
<th>Packet Number</th>
<th>Encrypted Payload...</th>
</tr>
</thead>
</table>
QUIC Packet

Cleartext Public Header

Flags | Connection ID | Packet Number | Encrypted Payload...

6
QUIC Packet

Does not depend on network 4-tuple

Cleartext Public Header

| Flags | Connection ID | Packet Number | Encrypted Payload... |
QUIC Packet

- **Flags**
- **Connection ID**
- **Packet Number**
- **Encrypted Payload...**

Does not depend on network 4-tuple

Monotonically Increasing

Cleartext Public Header
QUIC Packet

- Flags
- Connection ID
- Packet Number
- Encrypted Payload...

- Does not depend on network 4-tuple
- Monotonically Increasing
- No retransmission ambiguities

Cleartext Public Header
QUIC Packet

- Does not depend on network 4-tuple
- Monotonically Increasing
- Contains control/data frames
- No retransmission ambiguities

Cleartext Public Header

| Flags | Connection ID | Packet Number | Encrypted Payload... |
QUIC Packet

- **Flags**
- **Connection ID**
- **Packet Number**
- **Encrypted Payload...**

**Cleartext Public Header**

- Does not depend on network 4-tuple
- Monotonically Increasing
- Contains control/data frames
- No retransmission ambiguities
- Independent of packets

Slide 11
QUIC Packet

Flags | Connection ID | Packet Number | Encrypted Payload...

Cleartext Public Header

- Does not depend on network 4-tuple
- Monotonically Increasing
- Contains control/data frames
- No retransmission ambiguities
- Independent of packets
- Easier deployment
QUIC Data Transfer

STREAM(id=5,off=0):”Some data in my long frame”
QUIC Data Transfer

.byte 25, 5, 0

STREAM(id=5, off=0): "Some data in my long frame"
QUIC Data Transfer

F CID PN=25
STREAM(id=5,off=0):"Some data in my long frame"

F CID PN=19
ACK(25) MAX_DATA(for stream=5): 1024
QUIC Data Transfer

H1

F CID PN=25

STREAM(id=5,off=0): “Some data in my long frame”

H2

F CID PN=19

ACK(25) MAX_DATA(for stream=5): 1024

Control Frames
QUIC Data Transfer

**H1**

- **F CID PN=25**
  - **STREAM(id=5,off=0):"Some data in my long frame"**

**H2**

- **F CID PN=19**
  - **ACK(25)**
  - **MAX_DATA(for stream=5): 1024**

**H1**

- **F CID PN=26**
  - **STREAM(id=5,off=26):"."**
  - **STREAM(id=7,off=0):"Y"**
  - **ACK(19)**
QUIC Data Transfer

- **CID F PN=25**
  - STREAM(id=5,off=0):"Some data in my long frame"

- **CID F PN=19**
  - ACK(25)
  - MAX_DATA(for stream=5): 1024

- **CID F PN=26**
  - STREAM(id=5,off=26):”."
  - STREAM(id=7,off=0):”Y"
  - ACK(19)

Multiplexing
QUIC Data Transfer

H1

F  CID  PN=25  STREAM(id=5,off=0):"Some data in my long frame"

F  CID  PN=19  ACK(25)  MAX_DATA(for stream=5): 1024

F  CID  PN=26  STREAM(id=5,off=26):"."  STREAM(id=7,off=0):"Y"  ACK(19)

F  CID  PN=20  ACK(26)

H2
QUIC and Packet Losses
QUIC and Packet Losses

H1

| F | CID | PN=26 | STREAM(id=5,off=26):"." | STREAM(id=7,off=0):"Y" | ACK(19) |

H2


QUIC and Packet Losses

H1: F CID PN=26

STREAM(id=5,off=26):”.” STREAM(id=7,off=0):”Y” ACK(19)

H2: F CID PN=27

STREAM(id=5,off=26):”.” STREAM(id=7,off=0):”Y”
QUIC and Packet Losses

No Packet Number Reuse
QUIC and Packet Losses

H1

F  CID  PN=26  STREAM(id=5, off=26):"."  STREAM(id=7, off=0):"Y"  ACK(19)

No Packet Number Reuse

H2

F  CID  PN=27  STREAM(id=5, off=26):"."  STREAM(id=7, off=0):"Y"

F  CID  PN=20  ACK(Largest=27, Block=>25)
What about Multipath?
Why Multipath QUIC?

- QUIC assumes a single-path flow
Why Multipath QUIC?

- QUIC assumes a single-path flow
Why Multipath QUIC?

- QUIC assumes a single-path flow
Why Multipath QUIC?

• QUIC assumes a single-path flow

• Multipath QUIC
  - Bandwidth aggregation
  - Seamless network handover
    • Can try new WiFi while keeping using LTE
Design of Multipath QUIC

- Connection is composed of a set of paths
Design of Multipath QUIC

- Connection is composed of a set of paths
Design of Multipath QUIC

- Connection is composed of a set of paths

Performance monitoring?
Loss detection?
Path congestion control?
Design of Multipath QUIC

- Connection is composed of a set of paths
Design of Multipath QUIC

- Connection is composed of a set of paths

| Flags | Connection ID | Path ID | Packet Number | Encrypted Payload... |

Explicit path identification
Design of Multipath QUIC

- Connection is composed of a set of paths

Flags | Connection ID | Path ID | Packet Number | Encrypted Payload...

Explicit path identification | No path handshake
Design of Multipath QUIC

- Connection is composed of a set of paths

![Diagram showing multipath communication](image)

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID</th>
<th>Path ID</th>
<th>Packet Number</th>
<th>Encrypted Payload...</th>
</tr>
</thead>
</table>

Explicit path identification: **No path handshake**

Per-path numbering space
Multipath QUIC Data Transfer

- Path 1: WiFi
- Path 2: LTE
Multipath QUIC Data Transfer

Path 1: WiFi

Path 2: LTE
Multipath QUIC Data Transfer

Server via WiFi

Path 1: WiFi

Phone

Path 2: LTE

Server via LTE
Multipath QUIC Data Transfer

Path 1: WiFi

- Server via WiFi
  - F CID 1 PN=1 STR(id=5)
  - F CID 1 PN=1 STR(id=7,off=0)
  - F CID 1 PN=2 ACK(pid=1,1) ACK(pid=2,1)

Path 2: LTE

- Server via LTE
  - F CID 2 PN=1 STR(id=7,off=1024)
Multipath QUIC Data Transfer

Path 1: WiFi

- Server via WiFi
  - Frame: F CID 1 PN=1
  - STR(id=5)

- Phone
  - Frame: F CID 1 PN=1
  - STR(id=7, off=0)

  - Frame: F CID 1 PN=2
  - ACK(pid=1,1)

  - Frame: F CID 2 PN=1
  - STR(id=7, off=1024)

  - Frame: F CID 2 PN=1
  - ACK(pid=2,1)

Path 2: LTE

- Server via LTE
  - Frame: F CID 2 PN=1
  - STR(id=7, off=1024)

Frames not constrained to a particular path
Multipath Negotiation

CHELLO(MaxPathID=0x5)
Multipath Negotiation

H1

CHELLO(MaxPathID=0x5)

SHELLO(MaxPathID=0x3)

H2
Multipath Negotiation

Use up to 4 paths (0x0, 0x1, 0x2, 0x3)

CHELLO(MaxPathID=0x5)

SHELLO(MaxPathID=0x3)
Multipath Mechanisms

- Path management
- Packet scheduling
- Congestion control
Path Management

• How and when paths are established?

IP1  IP2  IP3  IP4
Path Management

• How and when paths are established?
Path Management

• How and when paths are established?
Path Management

• How and when paths are established?

• Fullmesh fashion
Path Management

• How and when paths are established?

• Fullmesh fashion

• ADD_ADDRESS + REMOVE_ADDRESS frames
Packet Scheduling

- Lowest-latency first

Diagram:
- 20 ms RTT
- 10 ms RTT
Packet Scheduling

- Lowest-latency first

![Diagram showing packet scheduling with RTT values]
Packet Scheduling

• Lowest-latency first

• What about when starting using a new path?
Packet Scheduling

• Lowest-latency first

• What about when starting using a new path?
Packet Scheduling

- Lowest-latency first

- What about when starting using a new path?

- Schedule all frames (not only STREAM)
Congestion Control

• **Multipath = need for coupled CC**
  - CUBIC would be unfair

• **Opportunistic Linked Increase Algorithm**
  - MPTCP state-of-the-art
How well does Multipath QUIC perform?
Evaluation of Multipath QUIC

- (Multipath) QUIC vs. (Multipath) TCP
  - Multipath QUIC: quic-go
  - Linux Multipath TCP v0.91 with default settings

- Mininet environment with 2 paths

![Diagram showing a client connecting to a server through two routers, one path labeled Path 1 and the other labeled Path 2.]
Evaluating Bandwidth Aggregation

- Download of 20 MB file
  - Over a single stream
  - Collect the transfer time
Evaluating Bandwidth Aggregation

- Download of 20 MB file
  - Over a single stream
  - Collect the transfer time
- For a loss-free scenario

20ms RTT, 20 Mbps

40ms RTT, 15 Mbps
Evaluating Bandwidth Aggregation

- **Download of 20 MB file**
  - Over a single stream
  - Collect the transfer time
- **For a loss-free scenario**
  - MPQUIC has 13% speedup compared to MPTCP
Evaluating Bandwidth Aggregation

- **Download of 20 MB file**
  - Over a single stream
  - Collect the transfer time

- **For a loss-free scenario**
  - MPQUIC has 13% speedup compared to MPTCP
    - MPQUIC less bursty than MPTCP
    - Probably due to CC skew on initial path in MPTCP
Evaluating Bandwidth Aggregation

- **Download of 20 MB file**
  - Over a single stream
  - Collect the transfer time

- **For a loss-free scenario**
  - MPQUIC has 13% speedup compared to MPTCP
    - MPQUIC less bursty than MPTCP
    - Probably due to CC skew on initial path in MPTCP

- **But what about other topologies?**
Evaluating Bandwidth Aggregation

- Experimental design, WSP algorithm
- 2x253 network scenarios
  - Vary the initial path
- Median over 15 runs

<table>
<thead>
<tr>
<th>Factor</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity [Mbps]</td>
<td>0.1</td>
<td>100</td>
</tr>
<tr>
<td>Round-Trip-Time [ms]</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>Queuing Delay [ms]</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>Random Loss [%]</td>
<td>0</td>
<td>2.5</td>
</tr>
</tbody>
</table>
Large File Download - No Loss

GET 20 MB, 506 simulations low-BDP-no-loss

Time TCP / QUIC

TCP better

QUIC better
Large File Download - No Loss

GET 20 MB, 506 simulations low-BDP-no-loss

Time TCP / QUIC

Single-path

TCP better

QUIC better

CDF

Time Ratio
Large File Download - No Loss

GET 20 MB, 506 simulations low-BDP-no-loss

Time TCP / QUIC
Time MPTCP / MPQUIC
Large File Download - No Loss

GET 20 MB, 506 simulations low-BDP-no-loss

MPQUIC better in 85% of cases
Large File Download - No Loss

GET 20 MB, 506 simulations low-BDP-no-loss

MPQUIC better in 85% of cases

Our extracted scenario
Large File Download - No Loss

GET 20 MB, 506 simulations low-BDP-no-loss

MPQUIC better in 85% of cases

Path 1: 49.4 ms RTT, 18.90 Mbps, 82 ms queuing delay
Path 2: 10.6 ms RTT, 0.43 Mbps, 11 ms queuing delay

Our extracted scenario

Path 1: 27.2 ms RTT, 0.14 Mbps, 34 ms queuing delay
Path 2: 46.4 ms RTT, 49.72 Mbps, 47 ms queuing delay
Large File Download - Losses

GET 20MB, 506 simulations, low-BDP-losses

CDF

Time TCP / QUIC
Time MPTCP / MPQUIC
Large File Download - Losses

GET 20MB, 506 simulations, low-BDP-losses

**QUIC copes better with losses**
Large File Download - Losses

GET 20MB, 506 simulations, low-BDP-losses

- **TCP SACK:** 2-3 blocks
- **QUIC SACK:** 256 blocks

**QUIC copes better with losses**
What is the actual benefit of Multipath to QUIC?
Actual Multipath Benefit

- **Experimental Aggregation Benefit**
  - Multipath QUIC/TCP vs. single-path QUIC/TCP

  - Zero goodput
  - MP gives 0 Mbps
  - 3 Mbps + 5 Mbps paths
    - MP gives 5 Mbps
  - 3 Mbps + 5 Mbps paths
    - MP gives 8 Mbps

- **Results depends on the first path used**
  - Handshake latency over initial path
Benefits of Multipath - No Loss

GET 20 MB, 253 scenarios low-BDP-no-loss

48%

77%

Exp. Aggregation Benefit

MPTCP vs. TCP

MPQUIC vs. QUIC

Protocol

Best path first
Worst path first
Benefits of Multipath - No Loss

% scenarios multipath has EAB >= 0, regardless of first path used

GET 20 MB, 253 scenarios low-BDP-no-loss

Exp. Aggregation Benefit

MPTCP vs. TCP

MPQUIC vs. QUIC

Best path first
Worst path first
Benefits of Multipath - Losses

GET 20 MB, 253 scenarios low-BDP-losses

- 32%
- 62%

Exp. Aggregation Benefit

MPTCP vs. TCP

MPQUIC vs. QUIC

Protocol

Best path first
Worst path first
What about congestion-prone networks?
### Experimental Design with High-BDP Networks

<table>
<thead>
<tr>
<th>Factor</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity [Mbps]</td>
<td>0.1</td>
<td>100</td>
</tr>
<tr>
<td>Round-Trip-Time [ms]</td>
<td>0</td>
<td>400</td>
</tr>
<tr>
<td>Queuing Delay [ms]</td>
<td>0</td>
<td>2000</td>
</tr>
<tr>
<td>Random Loss [%]</td>
<td>0</td>
<td>2.5</td>
</tr>
</tbody>
</table>
Multipath Benefits without Losses

GET 20 MB, 253 scenarios high-BDP-no-loss

Exp. Aggregation Benefit

20%

58%

MPTCP vs. TCP

MPQUIC vs. QUIC

Best path first
Worst path first
Completion Time Ratio with Losses

GET 20MB, 506 simulations, high-BDP-losses

CDF

Time Ratio

Time TCP / QUIC
Time MPTCP / MPQUIC
What about short transfers?
Short Transfer Evaluation with Low-BDP

• Download of a 256 KB file
  – Collect transfer time
• Median over 3 runs

<table>
<thead>
<tr>
<th>Factor</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity [Mbps]</td>
<td>0.1</td>
<td>100</td>
</tr>
<tr>
<td>Round-Trip-Time [ms]</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>Queuing Delay [ms]</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>
Comparison QUIC vs. TCP
Comparison QUIC vs. TCP

GET 256 KB, 506 simulations, low-BDP-no-loss

- Blue line: Time TCP / QUIC
- Orange line: Time MPTCP / MPQUIC

Shorter QUIC handshake
Multipath Not Really Useful...

GET 256 KB, 253 scenarios low-BDP-no-loss

- Exp. Aggregation Benefit
- 16%
- 5%

MPTCP vs. TCP
MPQUIC vs. QUIC

Protocol

Best path first
Worst path first
What about network handover?
Network Handover Support

- Apple MPTCP deployment mainly for handover
  - Main use case for Siri
Network Handover Support

- Apple MPTCP deployment mainly for handover
  - Main use case for Siri
- Request/Response traffic
  - 750 bytes request/responses
  - Measure delay seen by client

15ms RTT, 100% loss after 3 s
25ms RTT
Multipath TCP Handover

![Graph showing delay to answer request vs sent time for MPTCP]
Multipath TCP Handover
What Happened During MPTCP Handover?
What Happened During MPTCP Handover?
What Happened During MPTCP Handover?
What Happened During MPTCP Handover?

RTO

Req
What Happened During MPTCP Handover?
What Happened During MPTCP Handover?
What Happened During MPTCP Handover?
What Happened During MPTCP Handover?

Res

RTO

RTO

4G

Wi-Fi
What Happened During MPTCP Handover?

RTO

Res

RTO
And What About Multipath QUIC?
And What About Multipath QUIC?
What Happened During Handover?
What Happened During Handover?
What Happened During Handover?
What Happened During Handover?

RTO

FCID 2 PN  STR(Req)  PATHS(1 lossy)
What Happened During Handover?

RTO

FCID 2 PN

STR(Req) PATHS(1 lossy)
What Happened During Handover?

RTO

WiFi

4G

FCID 2 PN

STR(Res)
What Happened During Handover?
What about actual networks?
QUICTester Application

- Perform tests in actual networks
  - Does (MP)QUIC work in your networks?
  - Does MPQUIC provides better performances?
  - Application running on iOS11
    - https://itunes.apple.com/fr/app/quictester/id1322019644?mt=8
  - Feel free to provide feedback :-)
To sum up...
Conclusion

- **Multipath should be part of any transport protocol**
  - Most devices are multihomed
- **Designed and implemented Multipath QUIC**
  - Source code + artifacts + IETF draft available
  - See multipath-quic.org
- **Multipath more promising with QUIC than TCP**
  - Also opens potential new use cases