Connection ID Management
A.K.A. WHAT'S THIS THING CALLED AGAIN?
Development of Connection IDs

Sequence with Gaps (pre-PNE)

- Packet number gaps attempt to reduce correlation between CIDs
- Created HoLB – only allowed to skip CIDs if you’ve received them (and therefore know the gap)
- Really confusing to apply to multiple paths

<table>
<thead>
<tr>
<th>Seq.</th>
<th>CID</th>
<th>P#</th>
<th>Gap</th>
<th>Token</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>(A)</td>
<td>126</td>
<td></td>
<td>F(A)</td>
</tr>
<tr>
<td>0</td>
<td>(B)</td>
<td>23</td>
<td></td>
<td>F(B)</td>
</tr>
<tr>
<td>1</td>
<td>(C)</td>
<td>470</td>
<td></td>
<td>F(C)</td>
</tr>
<tr>
<td>2</td>
<td>(D)</td>
<td>9</td>
<td></td>
<td>F(D)</td>
</tr>
<tr>
<td>3</td>
<td>(E)</td>
<td>672</td>
<td></td>
<td>F(E)</td>
</tr>
</tbody>
</table>
Development of Connection IDs

<table>
<thead>
<tr>
<th>Seq.</th>
<th>CID</th>
<th>P# Gap</th>
<th>Token</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>(A)</td>
<td>126</td>
<td>F(A)</td>
</tr>
<tr>
<td>0</td>
<td>(B)</td>
<td>23</td>
<td>F(B)</td>
</tr>
<tr>
<td>1</td>
<td>(C)</td>
<td>470</td>
<td>F(C)</td>
</tr>
<tr>
<td>2</td>
<td>(D)</td>
<td>9</td>
<td>F(D)</td>
</tr>
<tr>
<td>3</td>
<td>(E)</td>
<td>672</td>
<td>F(E)</td>
</tr>
</tbody>
</table>

PNE: A, F(A)  
   B, F(B)  
   C, F(C)  
   D, F(D)  
   E, F(E)
Unordered Set (post-PNE)

- Fixes HoLB
- Easy to use on multiple paths
  - Just pick a different one!
- Requirement to change when peer changes difficult to reliably specify / implement
  - Did peer change by itself, so I need to change, or did they change because I changed?
# Development of Connection IDs

<table>
<thead>
<tr>
<th>Seq.</th>
<th>CID</th>
<th>Token</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>(A)</td>
<td>F(A)</td>
</tr>
<tr>
<td>0</td>
<td>(B)</td>
<td>F(B)</td>
</tr>
<tr>
<td>1</td>
<td>(C)</td>
<td>F(C)</td>
</tr>
<tr>
<td>2</td>
<td>(D)</td>
<td>F(D)</td>
</tr>
<tr>
<td>3</td>
<td>(E)</td>
<td>F(E)</td>
</tr>
</tbody>
</table>

-1, F(A)
0, F(B)
1, F(C)
2, F(D)
3, F(E)

A, F(A)
B, F(B)
C, F(C)
D, F(D)
E, F(E)
Development of Connection IDs

Sequence without Gaps (-13)

- No HoLB, because no packet number gaps
- Easier to specify behavior:
  - Use a higher sequence number than ever before when starting a new path
  - On each path, never use a sequence number less than the highest you’ve ever sent or received on that path

<table>
<thead>
<tr>
<th>Seq.</th>
<th>CID</th>
<th>Token</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>(A)</td>
<td>F(A)</td>
</tr>
<tr>
<td>0</td>
<td>(B)</td>
<td>F(B)</td>
</tr>
<tr>
<td>1</td>
<td>(C)</td>
<td>F(C)</td>
</tr>
<tr>
<td>2</td>
<td>(D)</td>
<td>F(D)</td>
</tr>
<tr>
<td>3</td>
<td>(E)</td>
<td>F(E)</td>
</tr>
</tbody>
</table>
Example

- Using ‘A’, ‘B’, ‘C’, etc. to represent CIDCs of increasing sequence number
- Actual sequence numbers will differ in each direction, but using ‘A’ in each direction here
- Multiple paths are hard to draw
Example

- Each side is using CID A
Each side is using CID A
- And also probing a side path with CID B
Example

- Each side is using CID A
  - And also probing a side path with CID B
  - The probe doesn’t affect what gets used on the main path
Each side is using CID A
- And also probing a side path with CID B
- The probe doesn’t affect what gets used on the main path
- Endpoint rolls forward to a new CID, C
Example

Each side is using CID A
- And also probing a side path with CID B
- The probe doesn’t affect what gets used on the main path
- Endpoint rolls forward to a new CID, C
  - The peer reciprocates
Example

- Each side is using CID A
  - And also probing a side path with CID B
  - The probe doesn’t affect what gets used on the main path
- Endpoint rolls forward to a new CID, C
  - The peer reciprocates
  - The CID change on the main path doesn’t affect what gets used on the probing path
Each side is using CID A
- Endpoint rolls forward to a new CID, C
- The peer rolls forward to a new CID, D
Example

- Each side is using CID A
- Endpoint rolls forward to a new CID, C
- The peer rolls forward to a new CID, D
  - ...but the endpoint never received D!
  - Rolls forward to E, the next available
Example

- Each side is using CID A
- Endpoint rolls forward to a new CID, C
- The peer rolls forward to a new CID, D
  - ...but the endpoint never received D!
  - Rolls forward to E, the next available
  - Peer rolls forward to E as well
Here be dragons....

NCID: B,C,D
Here be dragons....

Whoops, I’m out of CID's!

Just gave him three extras; that’s plenty.
It’s possible to become unclear whether a peer has actually used a CID you’ve issued

Given that, **how do I know when the peer needs more CIDs?**
Here be dragons....

Only packet with B disappears

Guess we're not using B...!
Here be dragons....

Only packet with B disappears

Sometime later, packet is delivered

SR with F(B)

Guess we’re not using B...!
Over a long-lived connection with many CIDs, it’s impractical to remember all CIDs ever associated with the connection
  - Potential memory exhaustion attack
  - Might require allocating load balancer state as well

But when is it safe to “forget” a CID?
  - Forget too early and peer can trigger a Stateless Reset by using a seemingly-valid CID

Circumstances where CIDs expire
  - CID with encrypted payload and key rotation
Proposal

- **NEED_CONNECTION_ID frame**
  - Analogous to BLOCKED, but use it before you are
  - Requests to have at least X CIDs beyond sequence number Y

- **RETIRE_CONNECTION_ID frame**
  - Declares an old CID no longer associated with this connection
    - Stop using **and stop recognizing the Stateless Reset Token** upon receipt
    - Sender can forget CID upon acknowledgement
  - Discuss: Need to retire individual CIDs or range of CIDs?
Here be “dragons”....

- Sequence number from end of handshake is currently “-1”
  - Negative numbers are annoying to some
- Server’s Preferred Address includes a CID for use in probing
  - Avoids waiting for a NEW_CID frame
  - …but what sequence number is that?
- Client’s CID from handshake doesn’t have a Stateless Reset Token

<table>
<thead>
<tr>
<th>Seq.</th>
<th>CID</th>
<th>Token</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>A</td>
<td>F(A)??</td>
</tr>
<tr>
<td>0</td>
<td>B</td>
<td>F(B)</td>
</tr>
<tr>
<td>1</td>
<td>C</td>
<td>F(C)</td>
</tr>
<tr>
<td>2</td>
<td>D</td>
<td>F(D)</td>
</tr>
<tr>
<td>3</td>
<td>E</td>
<td>F(E)</td>
</tr>
</tbody>
</table>