RPC-over-QUIC Kick-off

Document Strategy

Chuck Lever <chuck.lever@oracle.com>
A QUIC Primer

• QUIC is a secure connection-oriented network transport that runs over UDP, originally designed for web applications.

• The QUIC “streams” abstraction provides an ordered byte-stream service to applications. Streams can be unidirectional or bidirectional, and can be created by either endpoint, and there can be billions of streams per connection.

• A QUIC connection can migrate across multiple network paths. Connections have connection IDs that are independent of peer addresses.

• Confidentiality, peer and connection ID authentication, and endpoint address validation are built in.
Who Wants RPC-over-QUIC and Why?

• There could be significant functional overlap between RPC-with-TLS and RPC on QUIC

• We don’t yet have a clear answer to these questions
  • We do know that storage protocol implementers are already experimenting
  • But there are costs and benefits…
Potential Benefits of RPC over QUIC

• Separate streams for forward- and reverse- direction RPC transactions
• Fast recovery after network packet loss
  • Network path migration is transparent to RPC consumers
  • Advanced error and congestion detection and control (e.g., ECN)
• Transport headers and other metadata are deeply obscured
  • Also, no need for an RPC_AUTH_TLS probe
Challenges for RPC

• TLS is always on for now, introducing unwanted overhead in some cases
  
  • In fact, QUIC replaces the TLS record protocol, making it unsupported on the current class of offload NICs. One design goal for RPC-with-TLS was to be offload-enabled to reduce deployment costs
  
  • Most QUIC implementations are in user space, which does not efficiently serve kernel RPC consumers such as storage ULPs
No Expected Benefit

- RPC connections are typically long-lived, so 0-RTT reconnect is unlikely to be interesting for typical consumers of RPC such as NFS
- RPC record fragment framing is still necessary
QUIC-Specific Standards Action
An RPC-over-QUIC binding document

• RPC-related
  • RPC message framing on top of QUIC streams
  • An IANA request to assign appropriate netids
  • Multiple reliable and in-order flows per connection
    • Guidelines for RPC consumers that wish to utilize multiple flows
    • Update TI-RPC transport nomenclature
QUIC-Specific Standards Action
An RPC-over-QUIC binding document, continued

• QUIC-related
  • Guidelines for receivers to distinguish RPC-over-QUIC from RPC-over-UDP traffic and route QUIC connection IDs properly
  • Special requirements for utilizing QUIC’s Transport Layer Security
    • QUICv1 utilizes TLSv1.3 handshake
    • RPC-with-TLS ALPN and certificate usage guidelines apply
    • Always-on means some RPC-with-TLS security policies can’t be used
Proposed Standards Action
NFS on QUIC

• QUIC is in a class of network transport services that separate the connection abstraction from the flow/stream abstraction:

  • A QUIC *stream* is a reliable connection-oriented network transport that meets the suitability requirements outlined in RFC 8881 Section 2.9, but what about a QUIC *connection*?

• How does NFS (in particular, NFSv4 sessions) make use of multiple streams per connection? For example, what does BIND_CONN_TO_SESSION do in this world? Can each session slot use one stream?

• How is a server-dropped RPC transaction reported?
Proposed Standards Action
NFS on QUIC - Authentication

• QUIC is in a class of network transport services that manage peer authentication, formerly handled by RPCSEC GSS.

• Use of peer authentication material to authenticate EXCHANGE_ID and friends

• SECINFO (and MNT) will need to advertise the required TLS security level
WG Bureaucratic Actions

- Does this work fall under the existing Extension or Maintenance clauses, or is a charter update necessary?
- WG consensus to begin work on the proposed new document that specifies RPC-over-QUIC
- If approved, assignment of milestones and document authors
Bibliography

- RFC 8166 - RPC over an RDMA Transport
- RFC 9000 - The QUIC Transport Protocol
- RFC 9002 - Using TLS to Secure QUIC
- https://datatracker.ietf.org/doc/draft-ietf-nfsv4-rpc-tls/