Using NETCONF over QUIC connection

draft-ietf-netconf-over-quic-04

Jinyou Dai(Fiberhome/CICT) *(Remotely)*
Shaohua Yu(Pengcheng Laboratory)
Weiqiang Cheng(China Mobile)
Marc Blanchet(Viagenie)
Per Andersson(Cisco systems)

IETF 123, Madrid

Updates since Bangkok (IETF 122)

Forbid TLS early data ‘(section 3.1.1)

- The QUIC protocol uses TLS 1.3 messages to secure the transport. This means that Early data (aka 0-RTT data) is supported. [RFC9001] Early data (aka 0-RTT data) is a mechanism defined in TLS.

- As noted in Section 2.3 of [I-D.ietf-tls-rfc8446bis], the security properties for early data are weaker than those for subsequent TLS-protected data. In particular, early data is not forward secret, and there is no protection against the replay of early data between connections.

Updates since Bangkok (IETF 122)

- **Strike out unused transport protocols such as ‘BEEP’, ‘SOAP’ etc. from NETCONF protocol layer (figure 1 in section 4)**

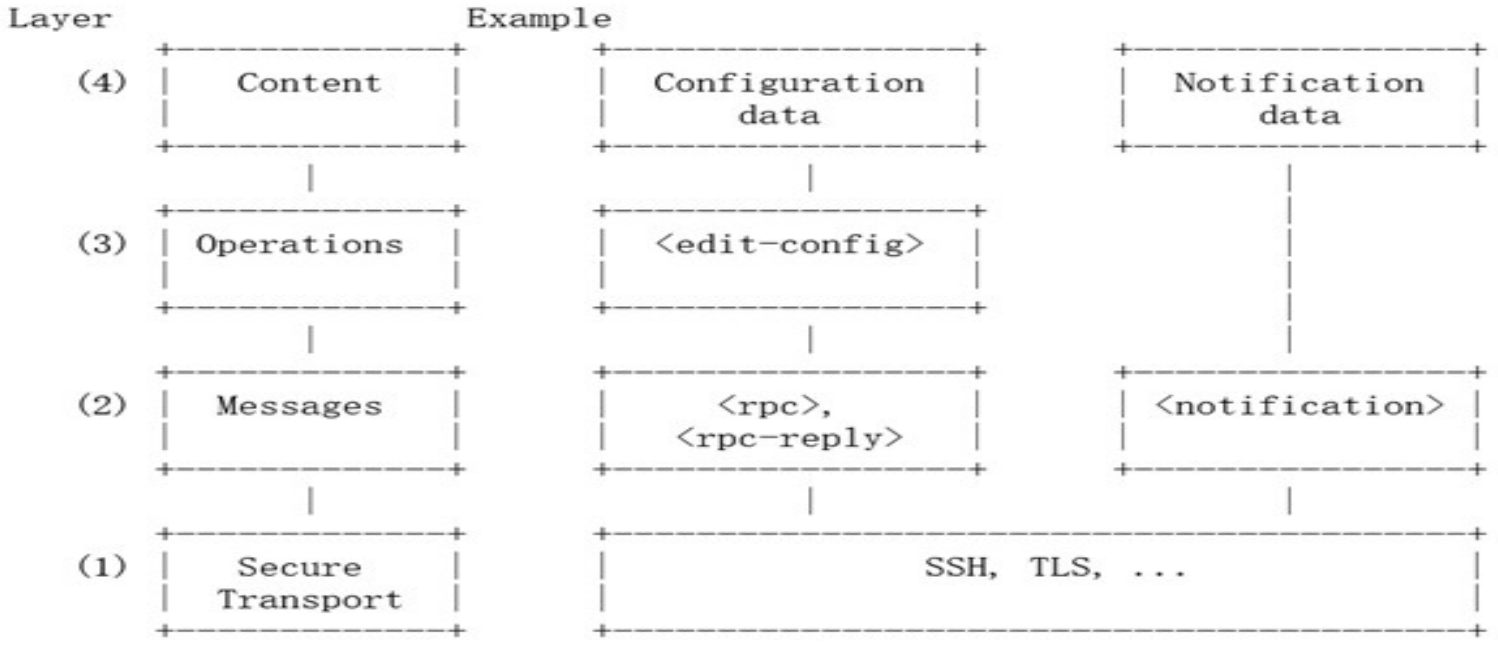

Figure 1: NETCONF Protocol Layers

Updates since Bangkok (IETF 122)

Modify ‘Bidirectional Stream Between client and server’(section 4.1)

- The messages used to exchange configuration data MUST be mapped into one or more bidirectional stream whose stream type is 0x0 according to the table 1.

- Since RPC processing is serialized and ordered within a session ([RFC6241] section 4.5), a bidirectional stream MUST be used for each NETCONF session.

Updates since Bangkok (IETF 122)

- **Add ‘RFC8071 Call Home Specific Case’ (in section 4.3)**
  - ✓ - Call home, RFC 8071
  - ✓ - NETCONF server initiates the connection
  - ✓ - Client, QUIC initiating party => NETCONF server
  - ✓ - Server, QUIC recieving party => NETCONF client

- **Elaborate ‘Security considerations’ (Section 8)**
  - ✓ The security considerations described throughout [RFC8446] and [RFC6241] apply here as well.
  - ✓ This document requires verification of server identity and client identity according to [RFC7589].

Remaining issues to be discussed

- mapping among NETCONF sessions, QUIC streams and QUIC connections
- Error code and stream error
- Security consideration
- About how to using stream types
- Nits fixing.

Updates for the next version

- **Improving description about NETCONF sessions, QUIC streams and QUIC connections**
  - ✓ Multiple NETCONF session is corresponding to one QUIC connection
  - ✓ One bidi-stream is corresponding to one NETCONF session.
  - ✓ Multiple uni-streams are corresponding to one NETCONF session.
  - ✓ One bidi-stream is corresponding to a RPC pairs.
  - ✓ One uni-stream is corresponding to a subscription.

- **Improving descriptions about error code.**
  - ✓ Register Application errors for CONNECTION_CLOSE frames and stream errors for RESET_STREAM and STOP_SENDING frames.

Updates for the next version

- **Improving description about security consideration**
  - ✓ Remove description about ‘an attacker inject arbitrary data into a QUIC stream’
  - ✓ Improve description about ‘The delimiter reference in Section 8’.
  - ✓ Elaborate description for citation of RFC 7589

- **processing description about stream types**
  - ✓ Option 1: keep table 1 and use stream type defined in RFC 9000 because backward compatibility can be assured by QUIC

  (backward compatibility is usually guaranteed, so, it is a proper solution to keep the original descriptions)
  - ✓ Option 2: Use the stream type name rather than stream type no.

- **Nits fixing and editorial improvement**

Next step

- Improve draft and address issues

- The authors would like to ask for WG last call after the publication of the -05 version

Thank you!