EPP over HTTPS (EoH)
and
EPP over QUIC (EoQ)
Implementation Experience

James Gould
jgould@verisign.com
IETF-121 REGEXT Working Group

Background

- EPP over HTTPS (EoH)
  - Defined in draft-loffredo-regext-epp-over-http
  - EoH connection maps to an HTTP session established with HTTP GET
  - EPP command sent over EoH connection with HTTP POST
  - Provides for Cloud-friendly EPP transport
- EPP over QUIC (EoQ)
  - Defined in draft-yao-regext-epp-quic
  - QUIC connection established with server that can create many QUIC streams
  - EoQ connection maps to a QUIC stream
  - EPP command sent over EoQ connection
  - Provides for secure and performant EPP transport

Implemented in Verisign EPP SDK

- Available at https://www.verisign.com/en_US/channel-resources/domain-registry-products/epp-sdks
- Java EPP SDK with client and server implementation
  - Support for many session pools
  - Support for 35 EPP extensions (object mappings and command response)
  - Stub server implementation to simulate server behavior
- Implemented and enhanced since the start of EPP

EoH Implementation Experience

- Technology
  - Client use of Java HttpClient, introduced in Java 11
    - Only supports HTTP/1.1 and HTTP/2
  - Server use of Embedded Tomcat 10 (10.1.30)
- Lessons Learned
  - EoH connection mapping to HTTP session and not HTTP connection
  - SSLContext helped reuse EPP over TCP (EoT) TLS configuration
  - CookieManager not set by default in HttpClient
  - Abstraction of HTTP protocol to support HTTP/1.1, HTTP/2, and HTTP/3 (future)
    - HTTP/3 support submitted for Java HttpClient but not implemented
  - HTTP headers needed changes in draft
    - Content-Type with charset matching EPP character encoding
    - Disable caching via “Cache-Control” and “Expires” headers
- Future
  - Test with HTTP/3 when supported by Java HttpClient and Embedded Tomcat

EoQ Implementation Experience

- Technology
  - Use of Kwik (version 0.8.13) for client and server
- Lessons Learned
  - Kwik still needs some work
    - Limited configurability (TLS, disable host name validation in Dev, logging)
    - Limited functionality, such as inability of retrieving client certificate in mTLS connection
  - Need to write data to create QUIC stream in server
    - Added EoQ Connection Start Packet to trigger EoQ connection and EPP Greeting
    - Server needs to accept stream prior to reading data
- Future
  - Use more than one QUIC stream per QUIC connection
  - Performance test for comparison with EPP over TCP (EoT)
  - Add retrieval of client certificate once Kwik supports it

EoH and EoQ Conclusion

- Fully compliant as EPP RFC 5730 transport defined in Section 2.1
- Fully pluggable with EPP over TCP (EoT) defined in RFC 5734
- Full suite of EPP tests ran on top of EoT, EoH, and EoQ
- Draft updates made based on implementation experience
- Encourage review and additional implementation experience
- Ready for REGEXT working group adoption