QUIC Version Aliasing

https://datatracker.ietf.org/doc/draft-duke-quic-version-aliasing/

First Connection

Client Initial, version 1

The salt and PLO are a secure hash

f(version, ITE)

(4233527)

Server Handshake, TP with: - random version number (0x433ad370)

Server Initial, version 1

 random Initial Token Extension(ITE) (0x19a25b)

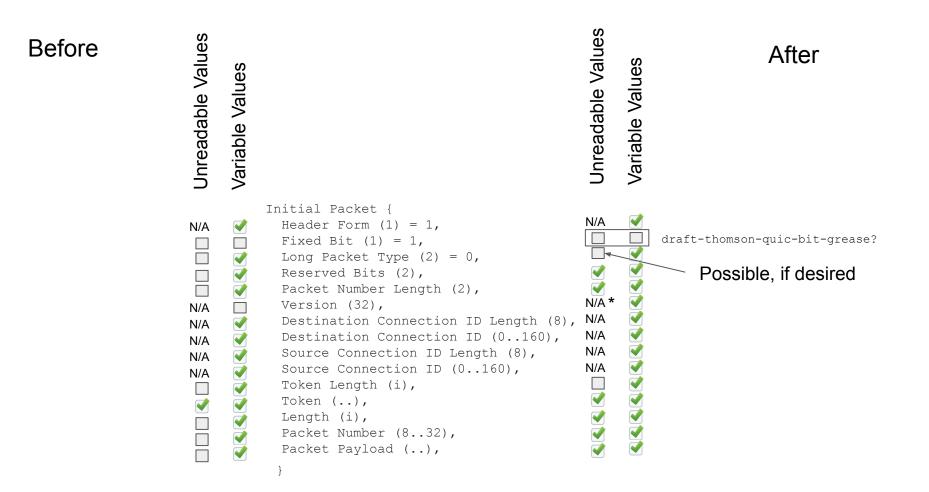
- salt (0x453acf30...) packet length offset

Next Connection

Client Initial, version 0x433ad370 token {N} + 0x19a25b length = 1200+4233527

Server computes salt from version, ITE

- Connection continues with aliased version number
- Server SHOULD issue TP with new values



Claimed Properties

- From second connection, Initial packet payloads are entirely private and immune from ossification
- Minimal TLS ossification vectors over QUIC
- Greases the version field
- Initial Injection attacks are over (maybe VN might work)
- Server has no per-client state
- More space-efficient than ECHO, covers the whole Initial packet, both authenticated and private in both directions
- Does nothing for the first connection
- Dependency on quic-version-negotiation
- Browsers & economically important websites need to deploy it to prevent firewalls from killing it

Feedback wanted:

https://github.com/martinduke/quic-version-aliasing

Any browsers and "economically important websites" interested?