QUIC Version Negotiation

draft-schinazi-quic-version-negotiation

IETF 106 – Singapore – 2019-11

David Schinazi – dschinazi@google.com
Eric Rescorla – ekr@rtfm.com

QUIC Version Negotiation already exists!

QUIC invariants define the Version Negotiation packet

QUIC transport instructs clients to fail the connection upon receiving it

Client could just reconnect with new version

- Downgrade attacks
- Spending 1-RTT to negotiate the best version is a dealbreaker for browsers

But we had downgrade prevention in the spec!

Original downgrade prevention mechanism relied on server sending supported versions and client validating that its previously attempted version wasn't there

But... breaks multi-server deployments with different versions
- multi-CDN and incremental rollout of new software

Decision at the time: remove from the spec and build later as extension

Here comes the extension, with 2 goals:
- downgrade prevention
- version negotiation without spending an RTT

Compatible Versions

If QUICv2 ends up being very similar to QUICv1,
it would be nice to use it without spending an RTT if the server doesn't support it

"Version A is compatible with version B" means:
the server can understand the client's first flight in version A
and can map it to a first flight in version B

Mechanism: client sends first flight in version A and indicates support for version B
- if server supports B, it responds with version B (as if client had started with B)
- otherwise responds with version A

A New Transport Parameter – Client

Currently Attempted Version
(adds version in long header to TLS key schedule)

Previously Attempted Version
(when VN was received)

Copy of Received Version Negotiation Packet Payload
(when VN was received)

Compatible Version List

A New Transport Parameter – Server

Negotiated Version

(adds version in long header to TLS key schedule)

Supported Version List

(allows client to save server's versions for future use)

Downgrade Prevention

Attacker tampers with the version in the client's first flight

- Currently Attempted Version won't match

Attacker tampers with the version in the server's first flight

- Negotiated Version won't match

Attacker forges a Version Negotiation packet (or tampers with real one)

- Server will notice it when verifying the client's Previously Attempted Version could generate the Copy of Received Version Negotiation Packet Payload

Next Steps

We've gotten good feedback on the list

Need to refactor the draft to make things clearer

Thoughts on WG adoption?

QUIC Version Negotiation

draft-schinazi-quic-version-negotiation

IETF 106 – Singapore – 2019-11

David Schinazi – dschinazi@google.com
Eric Rescorla – ekr@rtfm.com