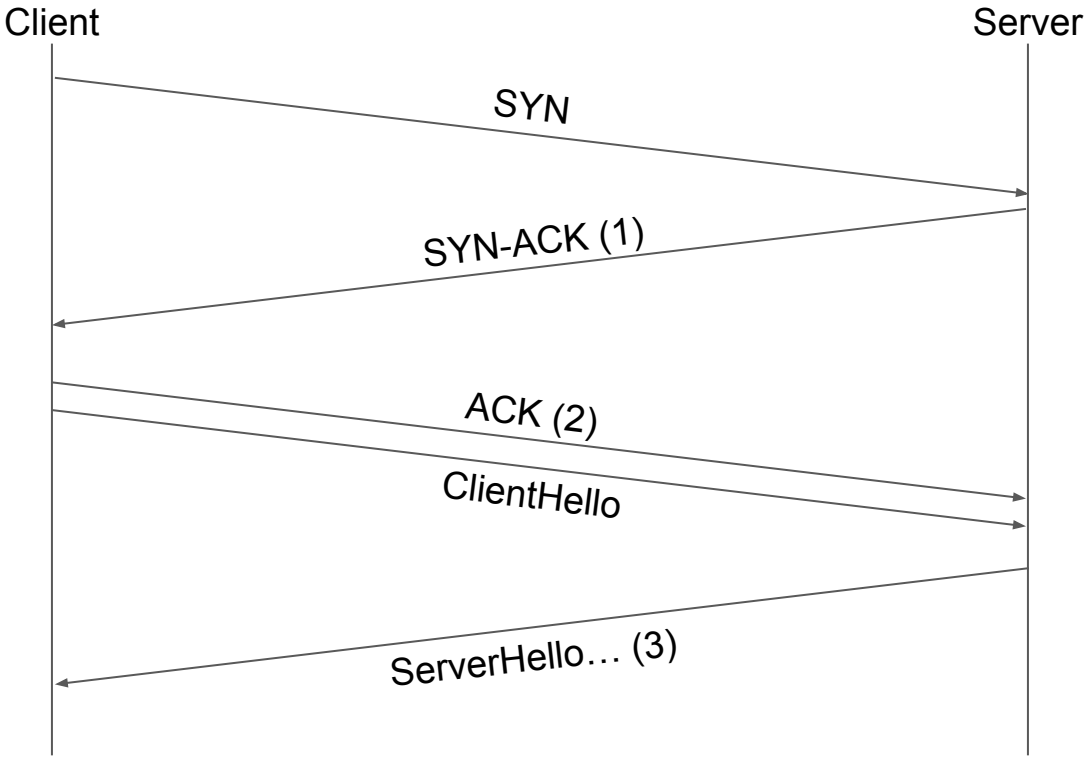
Certificate compression

draft-ghedini-tls-certificate-compression
IETF 98

Why certificate size matters

1. General network performance
2. Limiting QUIC amplification

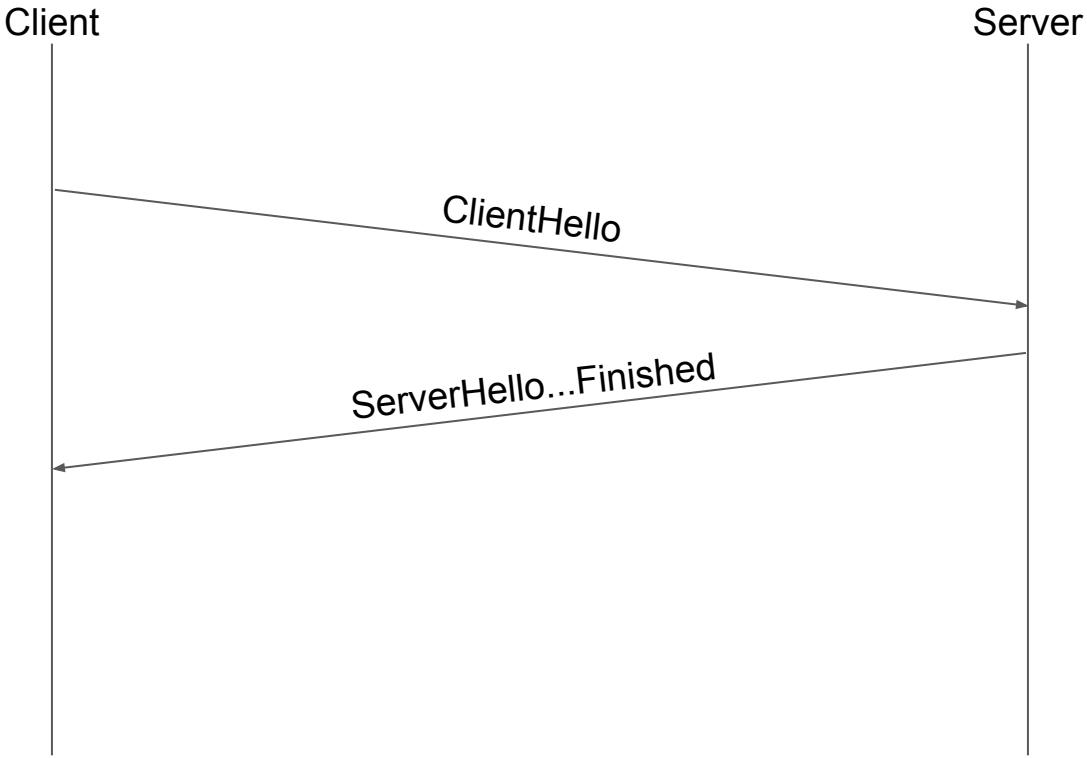
TLS first handshake: TLS 1.3 over TCP



In TLS over TCP, client proves ownership of IP address before sending ServerHello...Finished flight:

(1) Server issues a challenge as sequence number in SYN-ACK
(2) Client echoes it back in ACK
(3) ServerHello...Finished is sent after challenge succeeds

TLS first handshake: TLS 1.3 over QUIC



In TLS over QUIC, connection establishment and TLS handshake are combined. Hence, ServerHello...Finished can be used for UDP amplification attacks by spoofing IP addresses.

Solution: bound amplification by making flights smaller.

How does this work?

Use general-purpose compression, DEFLATE and Brotli.

Based on analysis of ~30k certificate chains from popular websites:

Compressing chains with Brotli yields (rough estimate):

- −30% size reduction at median
- −48% size reduction at 95th percentile
- Chains fitting into two QUIC packets: 2% → 54%
- Chains fitting into three QUIC packets: 55% → 97%

Why does this work?

What are leaf certificates actually made of:

- ~14% signatures
- ~15% keys
- ~14% SAN fields
- ~13% OIDs
- ~18% DER framing
- ~10% URLs
- ~12% other strings

cryptographic material (not compressible)

predictable and/or redundant content

Names in chains are inherently redundant.

Discussion