On the security of QUIC

Masaya Iseki[1], Eiichiro Fujisaki[2]
[1]Tokyo Institute of Technology ,[2]NTT Secure platform
e-mail : iseki.m.aa@m.titech.ac.jp, fujisaki.eiichiro@lab.ntt.co.jp

1 Introduction

Quick UDP Internet Connections (QUIC for short) is a new transport layer network protocol recently proposed by Google [?], which is experimentally implemented in Google Chrome. The main purpose of developing QUIC is to provide an alternative equivalence of TLS wrapping TCP, with much reduced latency and better SPDY support. Transport Layer Security (TLS) starts with a three-move TCP handshake before initiating the TLS Handshake Protocol. In contrast, QUIC uses UDP and starts with its own handshake, which reduces the total number of interactions. The cryptographic core of QUIC is specified in the QUIC crypto protocol [?], which consists of a handshake protocol and a record layer protocol, as does TLS. Similarly to TLS, QUIC has two types of handshake connections. One is called a full handshake – a handshake "from scratch" between a client and a server. The other is called a resumption – an abbreviated handshake, which occurs when a client and a server have once established a full handshake session and want to establish a new session between them in an abbreviate way. Unlike TLS, QUIC only supports the elliptic-curve Diffie-Hellman key-exchange (ECDHE) cipher-suite and server authentication. One of the good features of QUIC is that it can establish a resumption session with 0-RTT connectivity overhead. Namely, in the QUIC resumption, a client can send encrypted data to a server, concurrently with resuming a new session. We provide the abstract model of the full handshake and resumption protocols of QUIC in Fig. ??.

2 Prior Security Analyses and Some Security Concern

To the best of our knowledge, there are only two security analyses on QUIC [?, ?]. Both papers define new security models and show that QUIC is secure in that model. In [?], they formalized a secure authenticated key-exchange as an extension of the Bellare-Rogaway model [?] and analysed the security of QUIC (with resumption). However, the QUIC protocol analysed in [?] is slightly different from the protocol given in the source codes. As described in Fig. ??, the QUIC protocol makes a server send a ciphertext (using authenticated encryption) in the full handshake protocol, which cannot preserve *key-indistinguishability*. Therefore, the authenticated and channel confidentiality establishment (ACCE) model [?] is more suitable to analyse QUIC. Another important security issue is that in [?], an adversary is allowed to send a "test" query only to a client oracle (to receive either a real session-key or a random key from the client oracle), when a protocol is server-only authenticated. Apparently, the restriction is appropriate, because an adversary can establish a session with a honest server (due to the lack of client's certificate) to share a session key. However, if resumption is provided, we should consider the attack that, after a honest client and a honest server establish a full handshake session, an adversary might hijack a resumption session – it might impersonate the initial client and share a session key with the server. To protect the attack, we should allow an adversary to send test queries to *server* oracles in resumption sessions (including the full handshake session), as long as the initial full handshake session is established between a honest client and a honest server. We can consider a weaker attack: The adversary cannot share a session key with the server, but it can make the server accept in a resumption session. (Note that in a full handshake session, it is a "trivial" attack, because an adversary can always do so.) Protection of this attack guarantees that only parties that establish the initial full handshake session can resume and establish a new resumption session. Without this protection, we would possibly have an actual inconvenience. The aim of resumption is to resume a new session more rapidly than a full handshake session. In QUIC, it actually achieves 0-RTT. However, if this attack succeeds, the client and server will have inconsistent cache data and they cannot resume a new session, which means that they must establish a full handshake session again.

2.1 Related Work

There are a huge body of works on authenticated key exchange protocols (See [**?**] for survey). An important stream of research dates back to Bellare and Rogaway [**?**]. However, as mentioned above, the QUIC full handshake protocol does not satisfy key-indistinguishability as in the Bellare-Rogaway like model, because a server sends a ciphertext (using authenitcated encryption) in the full handshake protocol, as does TLS. TLS Handshake Protocol is recently analysed in various security models, e.g., [**?**, **?**, **?**, **?**, **?**, **?**]. Still, the security model for analysing a server-only authenticated connection of TLS, i.e., Server-Only Authenticated and Confidential Channel Establishment (SACCE) [**?**], does not capture our security issues. This is because, besides not treating the resumption originally, the SACCE security model only concerns, which is more essential, *server* authentication and a *client's* message confidentiality. However, these issues appear in (some kind of) *client* authentication and *server* confidentiality.

2.2 Our Results

To treat the security issues above, we introduce a new security model, what we call *Resumable* SACCE (RSACCE) security, where we consider a server's message confidentiality, as well as a client's message confidentiality, where an adversary is allowed to send an encryption query to a *server* (to break a server's message confidentiality) both in the full handshake session and its successor resumption sessions, as far as the server establishes the initial full handshake session with a *honest* client. We also provide a stronger model, called strong RSACCE security, where we ensure that a sever can establish a resumption session only with the *same* client as initially connected in the full handshake session. We require in that model forward secrecy among all independent sessions. For resumption to be effective, we compromise but still require some level of forward secrecy among related resumption sessions.

We analyse QUIC as it is, and prove that QUIC meets RSACCE security, but it does not meet the strong RSACCE one. We also analyse an optional version of QUIC with CETV, QUIC with an optional client encrypted tag value (CETV) mechanism, and show that it meets strong RSACCE security.

謝辞 ....
....

参考文献

[1] 著者 1, 著者 2, 論文タイトル, 雑誌名, 巻 (出版年), 開始頁–終了頁.

[2] 著者, 文献名, 出版社, 出版年.

[3] Author1, Author2 and Author3, Paper Title, Journal Name, Vol. (Year), $**$–$**$.

[4] Author, Book Title, Publisher, Year.

[5] Author1 and Author2, Paper Title, in: Proc. of Proceedings Name, Vol. $**$, pp. $**$–$**$, Year.

[6] JSIAM 2015 Official Page, `http://annual2015.jsiam.org/`.

※ 本文の出現順に並べてください.