# Robust and Scalable DTLS Session Establishment

by Marco Tiloca, Christian Gehrmann and Ludwig Seitz (SICS)

***The Datagram Transport Layer Security (DTLS) protocol is highly vulnerable to a form of denial-of-service attack (DoS), aimed at establishing a high number of invalid, half-open, secure sessions. Moreover, even when the efficient pre-shared key provisioning mode is considered, the key storage on the server side scales poorly with the number of clients. SICS Swedish ICT has designed a security architecture that efficiently addresses both issues without breaking the current standard.***

Secure communication is a main requirement in increasing numbers of applications, ranging from plant monitoring to home automation; from certified e-mail to e-commerce. Given the increasing number of applications relying on datagram protocols, the IETF has standardised the DTLS protocol [1], which is designed to be as similar as possible to the widely adopted TLS protocol.

Two DTLS peers, namely client and server, establish a secure session by performing a handshake. Typically, the client takes the initiative, by sending a ClientHello message to the server. During the handshake, the two peers establish and exchange the security material used later on. To this end, they may adopt an efficient key provisioning mode based on symmetric pre-shared keys (PSKs), so avoiding the computational complexity of public key cryptography and the management of a public key infrastructure. This has made PSK increasingly popular, as it is particularly suitable for applications like smart metering and building automation, where servers might be resource-constrained devices operating over low-bandwidth networks.

Nevertheless, the DTLS handshake displays two relevant security and performance issues.

Firstly, the server is highly vulnerable to a specific denial-of-service (DoS) attack. Specifically, an adversary can repeatedly send ClientHello messages to the server, and force it to start performing a considerable number of handshakes. While a preliminary Cookie exchange with the client can complicate the attack performance, it does not fundamentally protect the server against a DoS mounted by a determined and resourceful adversary. Hence, the server can still be induced to establish a considerable amount of half-open DTLS sessions, to exhaust its network resources and make it less responsive, or even unavailable, to legitimate clients.

Secondly, if the PSK provisioning mode is adopted, the server may have to store and manage a considerable number of pre-shared keys, possibly one for every possible client. This scales poorly with the number of clients and considerably complicates key provisioning operations, especially in dynamic environments.

SICS Swedish ICT [L1] has designed an efficient security architecture based on a Trust Anchor entity in a trusted relation with multiple DTLS servers. The architecture addresses the two identified DTLS issues, by combining the two following improvements.

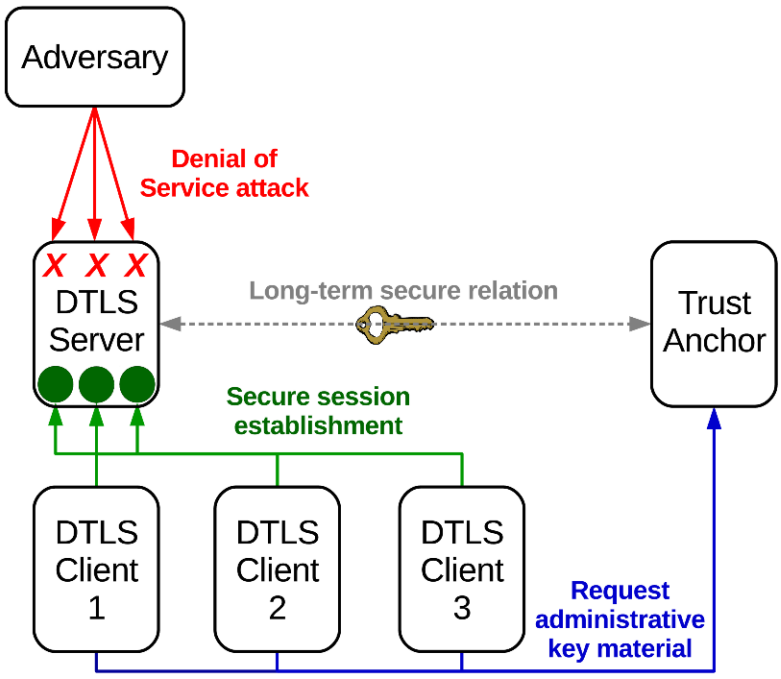


*Figure 1: Security architecture for robust and scalable DTLS session establishment.*

First, we have defined a preventive solution to the considered DoS attack. In particular, the server is able to identify invalid ClientHello messages and promptly abort the handshake execution at the first step, so practically neutralising the DoS attack and substantially limiting its impact. Besides, one message round trip between client and server can be avoided, as the cookie exchange is no longer necessary.

Second, we have defined an alternative PSK scheme that reduces the number of pre-shared keys stored by the server to one only, so preventing scalability and management issues and greatly reducing the load on the server. This is achieved by shifting the load of key management to the trust anchor and requiring clients to perform an additional message round trip.

Our approach displays a number of benefits. First, it relies on a standardised method to extend ClientHello messages, and does not require changes to the DTLS standard. Second, no additional message exchange between client and server is required, and even the cookie exchange is no longer necessary. Third, it does not significantly contribute to the computing overhead of client and server, as the handshake process maintains the same order of computational complexity. Finally, our improvements can also be easily readopted in the TLS protocol, without changing the actual standard.

We implemented our DTLS improvements in the library Scandium [L2] and performed an experimental performance evaluation. Results show that, when compared with the original DTLS protocol, our approach: i) improves a server's robustness against DoS; ii) reduces the time a server is exposed to a promptly neutralized DoS instance; and iii) improves service availability and scalability of key storage.

A comprehensive description of the architecture described above and the performance evaluation is available in [2].

Our approach is currently considered in the European project SEGRID [L3], which is devoted to improving cyber security in smart electricity grids. The project partners SICS Swedish ICT and the European Network for Cyber Security (ENCS) [L4] have been cooperating to integrate the presented DTLS improvements in a real substation automation system. This will contribute to make secure communication between RTU units based on DTLS and the IEC104 protocol more robust and scalable, hence more reliable and resilient. Final tests and performance evaluation will be performed in the project testbed SITE.

**Links:**
[L1] https://sics.se
[L2] https://github.com/mkovatsc/Scandium
[L3] http://www.segrid.eu
[L4] https://encs.eu

**References:**
[1] E. Rescorla and N. Modadugu: "RFC 6347, Datagram Transport Layer Security Version 1.2.", Internet Engineering Task Force, 2012.
[2] M. Tiloca, C. Gehrmann, L. Seitz: "On Improving Resistance to Denial of Service and Key Provisioning Scalability of the DTLS Handshake", International Journal of Information Security, Springer, 2016 (To appear)

**Please contact:**
Marco Tiloca
SICS Swedish ICT
marco@sics.se