DTLS In Constrained Environments (DICE)

The purpose of this document is to organize interest in a group to standardize a set of optimizations to DTLS for use with constrained devices and networks. Possible goals for this activity would be:

- A minimal configuration profile of DTLS for IoT
- Group communication security supported by DTLS Record Layer

Work that belongs to other groups, or possible future work:

- Implementation and deployment guidance (-> LWIG)
- Work on appropriate revocation for constrained devices (-> possible rechartering)
- TLS over CoAP handshake shim (-> possible rechartering)
- Use of DTLS for security bootstrapping and key management (-> own activity)

IETF Berlin BOF Agenda (1 hour!)
Location: Wednesday 15.10 - 16.10, Potsdam 3

Jabber, minutes etc.

1. Problem space (couple slides) (5 min) - Chairs
    - Scope we are trying to achieve now
    - Possible work for future charters (revocation, transport shims)
2. Presentation on possible solutions
    - Profiling (10 min) - Hannes Tschofenig
    - Record layer group communication use? (10 min) - Sandeep Kumar
3. Open mike (5-10 min)

4a. Is this a topic that the IETF should try to address?
4b. Is this a topic that the IETF should not address?
4c. Do you not understand the problem well enough?
5. Proposed charter discussion (10 min)
5a. Do people think this is a charter it makes sense to propose?
5b. Not propose?
5c. Don't know enough?
6. How many people are willing to edit documents, comment documents, implementing?

Possible BOF Questions & Answers

TODO, help from all welcome. Enter a question you think might arise, and the answer would be great too...

Q Why a new WG, why are these work items not being done within the TLS WG?
A We intend to reuse DTLS within the constrained environment without modifying DTLS. For profiling, we intend to identify a subset among the different choices/extensions available in DTLS, such that this subset fulfills both the security and resource-constraint requirements. For group communication, we intend to reuse the DTLS record layer as a secure bulk transport mechanism for group messages, with key management out-of-scope.

Q Will the IoT be securely interoperable with the Internet, or are we creating islands which cannot interoperate due to this security work?
A The proposals will not modify any of the protocols that run on the Internet but reuse them within the IoT with smart choices on the various options available. Any Internet device can interoperate with IoT if they can support these choices.

Q Why don't we use the IP multicast security for the group communication?
A We do not exclude the use of any IP multicast security. Our proposal is for the constrained devices which already have DTLS for unicast communication and has no additional code space for an additional security protocol. Our proposal helps such devices reuse the DTLS record layer as a bulk transport mechanism for the group communication assuming the group security associations are done out-of-band or as part of a general key-management protocol when defined.

Q Why don’t we include key distribution/management as part of this work?
A We think that designing a generic key distribution and key management scheme for IoT would be a better approach, compared to devising a specific key management approach for the group communication. A generic key management scheme can be used to bootstrap IoT devices to enable unicast, group communication, setting up security association between devices as well as updating cryptographic keys.

Q What is the difference between the Profiling work of DTLS in DICE and the minimal (D)TLS Profile defined in LWIG?
A (I thought we have answers documented somewhere?)

Q Why can't we address these issues in CoRE WG?
A CoRE WG is not currently chartered to look at all the security issues, and there is no other WGs that are chartered to look at security for IoT. As we do not plan to modify TLS, therefore TLS is not the right place either.

Q Why is source authentication out-of-scope for secure group communication?
A It is true that source authentication might be required for certain critical use cases. We think that public-key signature is the only viable approach to provide source authentication, one way to achieve this is via object security where the payload is signed by the sender. For most other use cases such as lighting control, triggering of events by sensors in an IoT network, an authenticated group message is sufficient.

Mailing List

https://www.ietf.org/mailman/listinfo/dtls-iot

Profiling Work Item Strawman

http://tools.ietf.org/html/draft-keoh-dtls-profile-iot-00

Group Communication Security Work Item Strawman

http://www.ietf.org/id/draft-keoh-dtls-multicast-security-00.txt

Existing work

http://www.ietf.org/id/draft-keoh-lwig-dtls-iot-01.txt
http://www.ietf.org/id/draft-hartke-core-codtls-02.txt
http://www.ietf.org/id/draft-tschofenig-lwig-tls-minimal-03.txt
http://www.ietf.org/id/draft-keoh-tls-multicast-security-00.txt

Draft Charter - DTLS In Constrained Environments (DICE)

There is an increased use of wireless control networks in city infrastructure, environmental monitoring, industrial automation, and building management systems.

These wireless control networks comprise many electronic devices, sensors and actuators that are connected to each other, and in most cases Internet connected, thus creating a trend towards Internet of Things (IoT). The CoRE working group has defined a framework for resource-oriented applications intended to run on constrained nodes and networks. This connects devices which are constrained by power, limited amount of code size and memory in a network with severe limits on throughput. The Constrained Application Protocol (CoAP) can be used to manipulate resources on a device in these environments secured by Datagram Transport Layer Security (DTLS).

Over the past few years, there have been many efforts to implement DTLS on embedded systems in order to support Internet of Things (IoT) applications. In fact, Transport Layer Security (TLS) and its datagram variant were both invented for use in the Internet-based web applications, and implementers face many challenges to deploy (D)TLS on IoT devices that are limited in memory resources (RAM, Flash), CPU and power. In particular, (D)TLS supports a wide range of security features and functionalities, some of these features are not necessarily required for IoT applications. One of the goals of DICE working group is to document the immediate problems that hinder the deployment of DTLS on embedded systems and proposes a DTLS profile for CoAP-based IoT applications based on well understood application use cases.

Group communication is an important feature in IoT applications as it can be effectively used to convey messages to a group of devices without requiring the sender to perform multiple time- and energy-consuming unicast transmissions, one for each group member. For example, in a building control management system, Heating, Ventilation and Air-Conditioning (HVAC) and lighting devices can be grouped according to the layout of the building, and control commands can be issued to a group of devices. Unsecured group communication for CNNs is enabled by using CoAP on top of IP-multicast. However, it must be secured as it is vulnerable to the usual attacks (eavesdropping, tampering, message forgery, replay, etc). DTLS has been chosen by CoRE to protect CoAP unicast communications, and it https://docs.google.com/document/d/1Gw00WXRgjJJQJMv9seVpuXuLtlDYKhHE-7pX4eDo3g8/edit?usp=sharinghttps://docs.google.com/document/d/1Gw00WXRgjJJQJMv9seVpuXuLtlDYKhHE-7pX4eDo3g8/edit?usp=sharingwhttps://docs.google.com/document/d/1Gw00WXRgjJJQJMv9seVpuXuLtlDYKhHE-7pX4eDo3g8/edit?usp=sharingohttps://docs.google.com/document/d/1Gw00WXRgjJJQJMv9seVpuXuLtlDYKhHE-7pX4eDo3g8/edit?usp=sharinghttps://docs.google.com/document/d/1Gw00WXRgjJJQJMv9seVpuXuLtlDYKhHE-7pX4eDo3g8/edit?usp=sharinguldhttps://docs.google.com/document/d/1Gw00WXRgjJJQJMv9seVpuXuLtlDYKhHE-7pX4eDo3g8/edit?usp=sharing https://docs.google.com/document/d/1Gw00WXRgjJJQJMv9seVpuXuLtlDYKhHE-7pX4eDo3g8/edit?usp=sharingbe beneficial if the same security protocol, i.e., DTLS Record Layer can be used to protect CoAP group communication as well without changing the existing DTLS state machine. The goal of the DICE working group is to ensure that DTLS is the obvious choice for protecting CoAP and other UDP based protocols for the Internet

of Things. Key management of group keys is however out of scope of this working group.

The current design of DTLS leads to fragmentation of DTLS handshake messages over the wireless link, in particular when Raw Public-key and Certificate modes are used. From the various implementation experiences reported in the LWIG working group, the complexity of re-transmission and re-ordering of DTLS handshake messages in constrained networks has resulted in a significantly increased code size and RAM. Additional reliability mechanisms for transporting DTLS handshake messages are required as they will ensure that handling of re-ordered messages needs to be done only once in a single place in the stack. This working group may also look at alternative TLS transports in cooperation with the TLS WG.

This WG combines expertise from both the IETF Application and Security areas in order to work out the appropriate use of DTLS for the Internet of Things. DICE will work closely with LWIG to understand the complexity and overhead issues of DTLS, and to investigate the performance issues of the DTLS handshake. Cooperation with the TLS WG will be necessary for all activities in DICE.

The scope of this WG is to define the following:

1. Document the problems with the DTLS handshake for IoT, and define a suitable profile of DTLS for an IoT architecture and use case that minimizes the complexity and overhead of DTLS for constrained devices. The set of DTLS extensions and modes to be supported will be defined.

2. Define the reuse of DTLS Record Layer for secure CoAP group communication in combination with a (out-of-band delivered) group key for select cipher suites. The DTLS state machine should not be modified/altered and key management is outside the scope.

Goals and Milestones

Oct 2013 WG document for DTLS for Constrained Environments profile
Nov 2013 WG document for secure COAP group communication for IoT
Feb 2014 DTLS for IoT profile specification submitted to the IESG for publication as standards track
Mar 2014 Secure COAP group communication specification submitted to the IESG for publication as standards track

Initial brainstorming

A minimal configuration profile of DTLS

[Klaus: Implementations in a deployment need to agree on cipher suite, etc. for interop. Maybe it's possible to define minimal profiles for a number of typical scenarios?]

[Sandeep: when selecting the minimal profiles for different typical scenarios it would be nice to select a common underlying cipher (like AES) to reduce implementation costs, so using AES-CMAC for authentication and AES-CCM for additional confidentiality

Zach: Common Ciphers are already defined in the CoAP specification.]

What defines a "profile"?

Maybe what we really need from a profile is to explain the (D)TLS features actually needed for a set of IoT applications that we commonly come across. The purpose of this profile would be to limit the number of options and flexibility in especially the (D)TLS handshake. What are the features actually needed for use with CoAP?

- Applicability
    - Communication model
    - Threat model
    - Security requirements
    - Class of devices
    - ...
- Cipher suites
- Extensions
    - Signature Algorithms
    - Server Name Indication
    - Maximum Fragment Length
    - Supported Elliptic Curves
    - Supported Point Formats
    - Application Layer Protocol
    - Heartbeat
    - Cached Info
    - Session Resumption
    - ...

- Timer values
- ...

[Note RS: it would be good to define technical requirements first here.
-security services (authenticated key agreement, authenticated key transport, entity authentication, secured data transport (unicast, multicast, broadcast; source authentication, etc.), replay protection, timeliness, etc. Authorization (identity-based, role based), privacy, etc.
-communication services: in-order delivery, fragmentation/defragmentation support, etc.
-support for initial provisioning, configuration
-scalability, adaptability towards change of role/trust models, device replacement, merging and partitioning of networks, flexible authorization policy management, etc.
- how to deal with typical network aspects, including sleepy nodes, crappy links, no online availability trusted third party
-how to make sure human involvement is virtually absent
-support for existing/legacy implementation pieces (e.g., AES in hardware; RNG, etc., device id)
-support for memory-starved devices, with harldy any buffering capability, etc.

Zach: Well, we have already been through the main requirements when we chose DTLS as the solution for CoAP in general. Much of this is also captured in the main CoAP ID. However for the purpose of optimizing DTLS, I agree we should capture the security requirements in relation to what we want to achieve. The goal here however is NOT to reconsider CoAP's choice of security protocol, model, cipher suites etc. The goal here is to make the choice we already made more efficient, and to support multicast.
]

Implementation and deployment guidance

Recommendations on

- handshake fragmentation to minimize retransmissions
- timer values to avoid spurious retransmissions
- when to perform handshake (on device startup, on demand, ...)
- when to close connections (when idle, …)
- DTLS version negotiation
- epochs >= 2
- session resumption

Dealing with resource exhaustion (eviction strategy for connections, ...)
Implementation techniques for implementing CoAP over DTLS
...

[Zach: This should be done in LWIG. Someone could already start writing this document up there?]

Optimization of the DTLS handshake and record layer

DTLS is written as a diff to TLS. The diff changes the transport to an unreliable datagram-based transport, adds reliability to the handshake layer and makes a few modifications to the record layer to cater for message reordering, etc.

DTLS was clearly not designed with constrained devices and lossy networks in mind. It should be possible to investigate alternate reliability mechanisms and message formats that are more suitable for constrained environments but do not touch TLS itself. This means basically reinventing the "D" in "DTLS", to a greater or smaller extent.

Ideas floating around:

- Transport the handshake messages in CoAP requests and responses, using draft-ietf-core-block to transport messages that do not fit in a single datagram.

- Transport the DTLS records in CoAP messages. For experimentation, I have added a tiny (5 bits) fragmentation indication to the CoAP messaging layer. The result can be used to transport the DTLS messages of a handshake flight in the same simple lock-step fashion as we already do in CoAP, instead of transmitting all messages at once and retransmitting all fragment when a single fragment is lost.

- Retransmit all fragments when a single fragment is lost and focus on reducing the number of fragments to be sent. Retransmitting all fragments actually performs better(!) than sending one acknowledgement per fragment in lossy environments in terms of transmission count.

[Note RS: it is not a priori clear that DTLS fits the bill here: first discuss requirements, instead of reverse engineering those from dreamt up solution that may be incomplete at best (I am not saying it is, but it could

Zach: This group is not about re-thinking security from scratch. We already chose (D)TLS as our security binding for CoAP. Based on experience using it so far, there is some obvious optimization that can be done, as well as an important feature missing.]

Multicast record layer support for DTLS

The current DTLS record layer header can be used to support/protect a single sender-multiple receivers group communication scenario. The sequence number field in

the record layer header is probably sufficient to detect replay attacks. This is based on the assumption that authorized devices in a multicast group are trusted to behave correctly. The question then is whether this assumption is reasonable and do we need additional measures to prevent insider attacks?

There was also a question whether source authentication is necessarily for multicast? In our opinion, this can be an optional feature in that for some important use cases, e.g., switching off all lights in the building, source authentication is required. Providing source authentication with a group key is not feasible, an alternative is to provide source authentication at the CoAP level using public-key cryptography?

A single security protocol for IoT

To strive for a single security protocol (e.g., the optimized "D"TLS) that can provide the required security functionalities such as network access, key management (distribution of unicast, multicast keys) and secure group communication such that we can optimize the use of memory on the constrained devices.

- Define the workflow in to perform network access using DTLS
- Define ways of distributing and renewing application, network, and multicast keys over the DTLS secure channel, i.e., using CoAP messages to encapsulate keying materials.
- Define the key derivation methods, re-using the security components in DTLS, e.g., PRF functions, AES.