14. RFC 7925： Transport Layer Security (TLS) / Datagram Transport Layer Security (DTLS) Profiles for the Internet of Things
RFC 7925：物聯網的傳輸層安全性（TLS）/ 資料封包傳輸層安全性（DTLS）配置文件

網際網路工程任務組(IETF)
Request for Comments：7925
分類：標準
ISSN：2070-1721

H. Tschofenig Ed.
ARM Ltd.
T. Fossati
諾基亞

2016 年 7 月

物聯網的傳輸層安全性（TLS）/ 資料封包傳輸層安全性（DTLS）配置文件

摘要

物聯網（IoT）部署中的常見設計模式是使用受限制設備，該設備通過傳感器或控制執行器收集數據，以用於家庭自動化，工業控制系統，智慧城市和其他物聯網部署。

本文定義了傳輸層安全性(TLS)和資料封包傳輸層安全性(DTLS) 1.2 配置文件，為此數據交換提供通信安全性，從而防止竊聽，篡改和消息偽造。缺乏通信安全性是物聯網產品中的常見漏洞，可以通過使用這些經過充分研究和廣泛部署的網際網路安全協定輕鬆解決。

本文的狀態

這是網際網路標準協定文件。

本文是網際網路工程任務組（IETF）的產品。 它代表了 IETF 社群的共識。它已經過公眾審查，並已獲得網際網路工程指導小組（IESG）的批准發布。 有關網際網路標準的更多訊息，請參見 RFC 7841 的第 2 節。

有關本文當前狀態，任何勘誤以及如何提供反饋的訊息，請訪問 http://www.rfc-editor.org/info/rfc7925。

版權聲明

目錄

1. 前言

開發物聯網（IoT）設備的工程師需要調查安全威脅並決定可用於緩解這些威脅的安全服務。

使 IoT 設備能夠交換數據通常需要對兩個端點進行身份驗證，並且能夠為交換的數據提供完整性和機密性保護。雖然可以在協定

疊的不同層提供這些安全服務，但是使用傳輸層安全性（TLS）/資料封包傳輸層安全性（DTLS）已經非常受許多應用程序協定的歡迎，並且它可能對物聯網很有用。

將網際網路協定安裝到受限制設備中可能很困難，但由於標準化工作，可以使用新的配置文件和協定，例如受限制的應用協定（CoAP）[RFC7252]。 CoAP 消息主要通過 UDP / DTLS 傳輸，但也可以使用其他傳輸，例如 SMS（如附錄 A 中所述）或 TCP（當前通過[COAP-TCP-TLS]提出）。

雖然本文的主要目標是使用 DTLS 1.2 [RFC6347]保護 CoAP 消息，但以下各節中包含的訊息不僅限於 CoAP，也不限於 DTLS 本身。

相反地，本文定義了 DTLS 1.2 [RFC6347]和 TLS 1.2 [RFC5246]的配置文件，它為物聯網應用程序提供通信安全服務，並且可在許多受限制設備上合理地實現。 因此，配置文件意味著利用可用的配置選項和協定擴充來最好地支援 IoT 環境。 本文不會改變 TLS / DTLS 規範，也不會引入任何新的 TLS / DTLS 擴充。

本文的主要目標讀者是配置和使用 TLS / DTLS 堆疊的嵌入式系統開發人員。但是，本文也可以幫助那些為物聯網產品開發或選擇合適的 TLS / DTLS 堆疊的人。 如果您熟悉（D）TLS，請跳至第 4 部分。

2. 術語

關鍵詞“必須(MUST)”，“絕不(MUST NOT)”，必要(REQUIRED)”，“應該(SHALL)”，“不應該(SHALL NOT)”，“應該(SHOULD)”，“不應該(SHOULD NOT)”，“推薦(RECOMMENDED)”，“不推薦(NOT RECOMMEMDED)”，“可能(MAY)”和“可選(OPTIONAL)”“在本文中將按照 RFC 2119 [RFC2119]中的描述進行解釋。

此規範涉及 TLS 以及 DTLS，特別是 1.2 版本，這是撰寫本文時的最新版本。

每當文本適用於協定的兩個版本以及兩個協定之間存在差異時的 TLS 或 DTLS 時，我們都會引用 TLS / DTLS。 請注意，正在

開發 TLS 1.3，但由於版本 1.3 的 TLS 發生了重大變化，因此預計此配置文件不會“正常工作”。

請注意，本文中的“客戶端”和“服務器”指的是客戶端啟動交握的 TLS / DTLS 角色。這不會限制 DTLS 之上的協定的交互模式，因為記錄層允許雙向通信。這方面將在 3.2 節中進一步描述。

RFC 7228 [RFC7228]引入了約束節點網路的概念，約束節點網路由對功率、內存和處理資源有嚴格限制的小型設備組成。術語受約束設備和 IoT 設備可互換使用。

術語“憑證頒發機構”（CA）和“專有名稱”（DN）取自[RFC5280]。術語“信任鏈”和“信任鏈儲存”在[RFC6024]中定義為：

> 信任鏈通過公鑰和關聯數據表示權威實體。公鑰用於驗證數位簽名，並且相關聯的數據用於約束信任鏈具有權威性的訊息類型。
>
> 信任鏈儲存是儲存在設備中的一組一個或多個信任鏈....設備可以具有多於一個信任鏈儲存，每個信任鏈儲存可以由一個或多個應用程序使用。

3. 概述

3.1. TLS 和 DTLS

TLS 協定[RFC5246]在兩個端點之間提供經過身份驗證，機密性和完整性保護的通信。該協定由兩層組成：記錄協定和交握協定。在最低級別，在可靠傳輸協定（例如，TCP）之上分層是記錄協定。它通過使用對稱加密技術來提供連接安全性，以實現機密性，數據源身份驗證和完整性保護。記錄協定用於封裝各種更高級別的協定。交握協定由三個子協定組成 - 即交握協定，更改密碼規範協定和警報協定。交握協定允許服務器和客戶端相互認證，並在應用協定發送或接收數據之前協商加密算法和加密密鑰。

DTLS [RFC6347]的設計有意與 TLS 非常相似。但是，由於 DTLS 在不可靠的資料封包傳輸之上運行，因此它必須明確地應對 TLS

所做的可靠和有序傳遞假設的缺失。 RFC 6347 非常詳細地解釋了這些差異。作為簡短的總結，對於那些不熟悉 DTLS 的人來說，不同之處在於：

- 記錄協定中包含顯式序列號和紀元欄位。 RFC 6347 的 4.1 節解釋了這兩個新欄位的處理規則。 用於計算消息認證碼（MAC）的值是通過連接時期和序列號形成的 64 位值。

- 流密碼不得與 DTLS 一起使用。 為 TLS 1.2 定義的唯一流密碼是 RC4，由於加密弱點，即使與 TLS [RFC7465]一起使用也不建議使用。注意，術語“流密碼”是 TLS 規範中的技術術語。RFC 5246 的 4.7 節定義了 TLS 中的流密碼，如下所示：“在流密碼加密中，明文與從密碼安全密鑰偽隨機數生成器生成的相同數量的輸出進行異或運算。”

- TLS 交握協定已得到增強，包括用於拒絕服務（DoS）抵抗的無狀態 cookie 交換。 為此，在 DTLS 中添加了新的交握消息 HelloVerifyRequest。 此交握消息由服務器發送，並包含無狀態 cookie，該 cookie 在 ClientHello 消息中返回給服務器。 儘管交換對於服務器執行是可選的，但是必須準備客戶端實現以響應它。此外，已經擴充了交握消息格式以處理消息丟失，重新排序和分段。

3.2. 通訊模型

本文描述了 DTLS 的概要，並且有用的是，它必須對預想的通信體系結構做出假設。

本文中描述了兩種通信體系結構（以及因此產生的兩種輪廓）。

3.2.1. 受限制的 TLS / DTLS 客戶端

圖 1 中所示的通信體系結構假定與使用與一個或多個 TLS / DTLS 服務器交互的受約束 TLS / DTLS 客戶端的 IoT 設備進行單播通信交互。

在客戶端可以啟動 TLS / DTLS 交握之前，它需要知道該服務器的 IP 地址以及要使用的憑據。 在 DTLS 之上傳送的應用層協定（例如 CoAP）可以配置有 CoAP 需要註冊和發布數據的端點的 URI。 該配置訊息（包括非機密憑證，如憑證）可以作為韌體/軟體包的一部分或通過配置協定傳達給客戶端。此配置文件支援以下憑據類型：

- 對基於預共享密鑰（PSK）的驗證（參見第 4.2 節），這包括配對的“PSK 標識”和要與每個服務器一起使用的共享密鑰。

- 對基於原始公鑰的身份驗證（請參閱第 4.3 節），這包括服務器的公鑰或服務器公鑰的 hash 值。

- 對基於憑證的身份驗證（請參閱第 4.4 節），這包括預先填充的信任鏈儲存，該儲存允許 DTLS 堆疊對與服務器交握期間獲得的憑證執行路徑驗證。

圖 1 顯示了儲存在受約束客戶端以供各個服務器使用的示例配置訊息。

本文重點介紹 DTLS 客戶端功能，但很自然地，必須提供等效的服務器端支援。

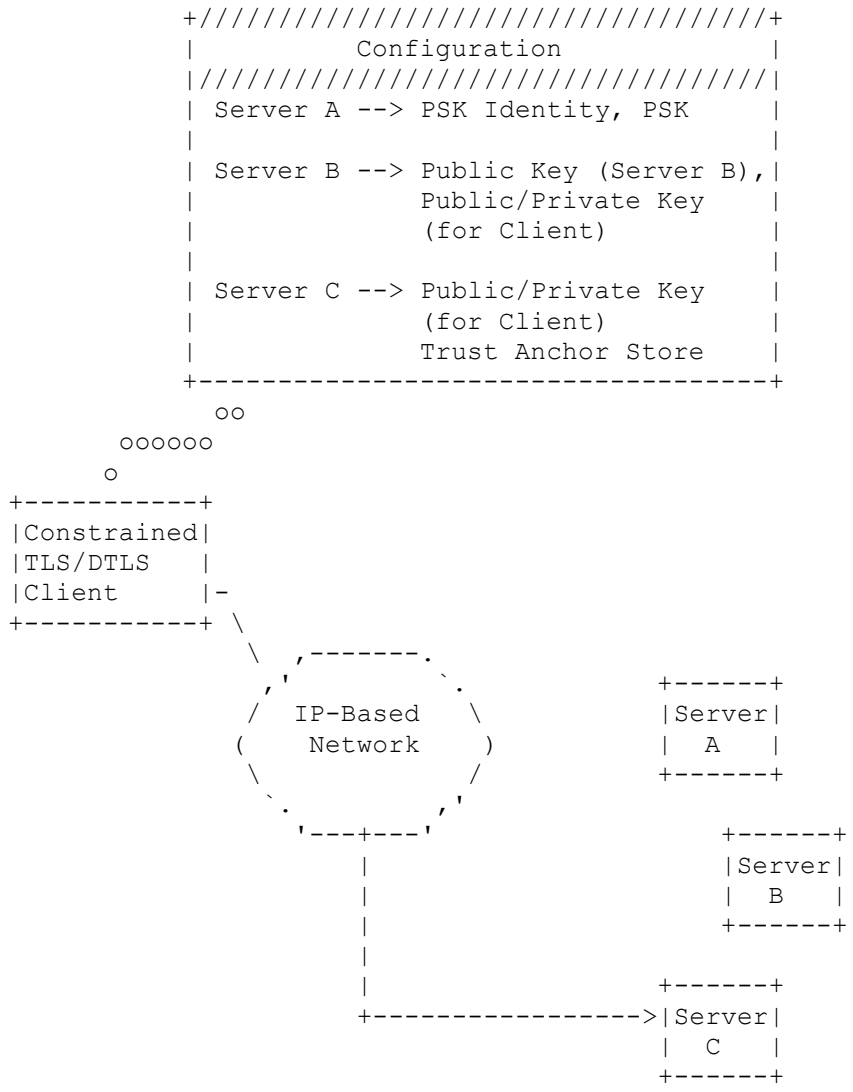


圖 1：約束客戶端配置文件

3.2.1.1. 約束客戶端交換的示例

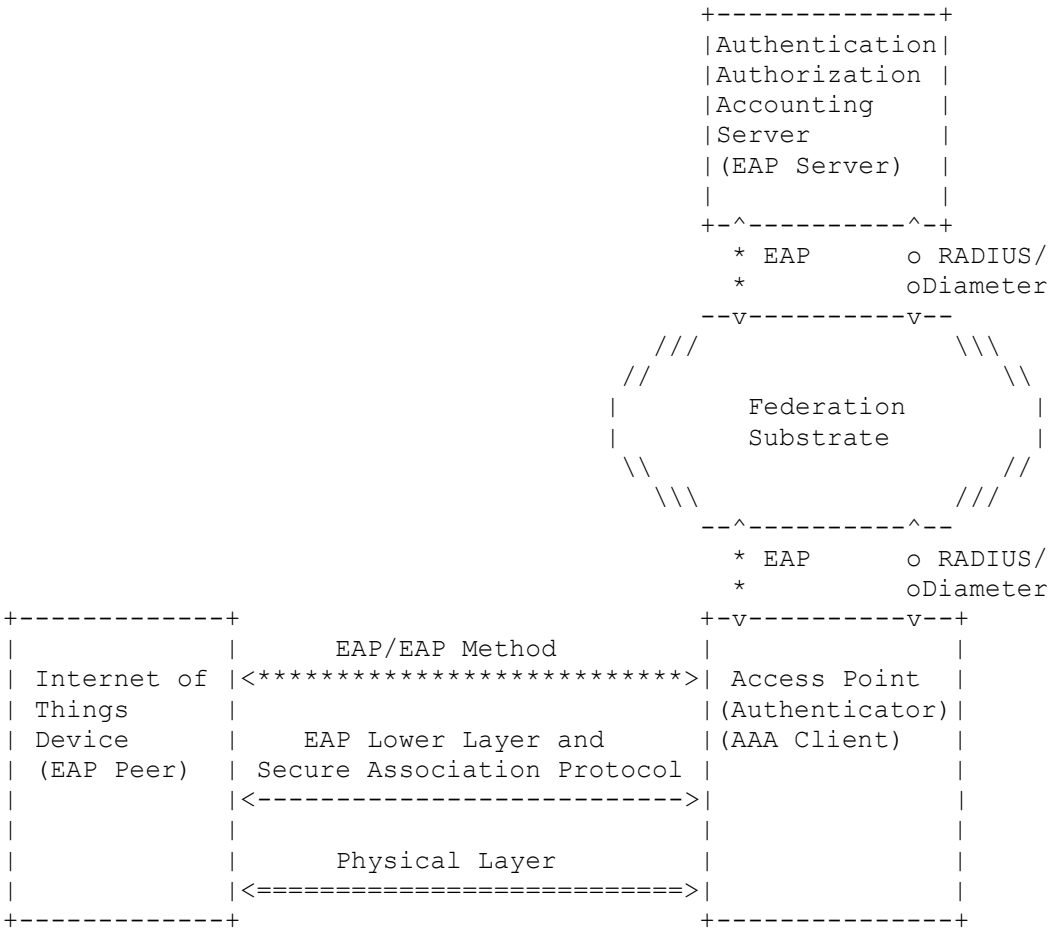
3.2.1.1.1. 網路訪問驗證示例

在考慮受約束的環境時，重用是一個反復出現的主題，並且是在受限環境的開發中採用的許多方向的背後。重用的必然結果是如果可以避免則不添加功能。 與使用 TLS 相關的示例是網路訪問認證，其在設備連接到網路並且在允許其與其他設備通信或連接到網際網路之前需要經歷認證和訪問控製過程時發生。

圖 2 顯示了網路訪問體系結構，其中 IoT 設備使用為特定物理層定義的過程啟動與網路中的接入點的通信。由於憑證可以集中管理和儲存，在認證，授權和計費（AAA）服務器中，安全協定交

換可能需要通過認證器中繼，即在接入點上運行的功能到 AAA 服務器。身份驗證和密鑰交換協定本身封裝在容器內，即可擴充身份驗證協定（EAP）[RFC3748]，並且消息在 EAP 端點之間來回傳遞，即位於 IoT 設備上的 EAP 對等端和位於 EAP 服務器上的 EAP 端點 在 AAA 服務器或接入點上。將來自作為 AAA 客戶端的接入點的 EAP 消息路由到 AAA 服務器需要適當的協定機制，即 RADIUS [RFC2865]或 Diameter [RFC6733]。

有關這些概念的更多細節和有關術語的描述可以在 RFC 5247 [RFC5247]中找到。



Legend:

<****>：設備到 AAA 服務器的交換
<---->：設備到身份驗證器交換
<oooo>：AAA 客戶端到 AAA 服務器交換
<====>：實體層，如 IEEE 802.11 / 802.15.4

圖 2：網路訪問體系結構

一種標準化的 EAP 方法是 EAP-TLS，在 RFC 5216 [RFC5216]中定義，它重用基於 TLS 的協定交換，將其封裝在 EAP 有效負載內。在重用方面，這允許在網路訪問安全功能和保護應用層流量所需的 TLS 功能之間共享 TLS 協定的許多組件。 在圖 3 所示的 EAP-TLS 交換中，作為 EAP 對等體的 IoT 設備充當 TLS 客戶端。

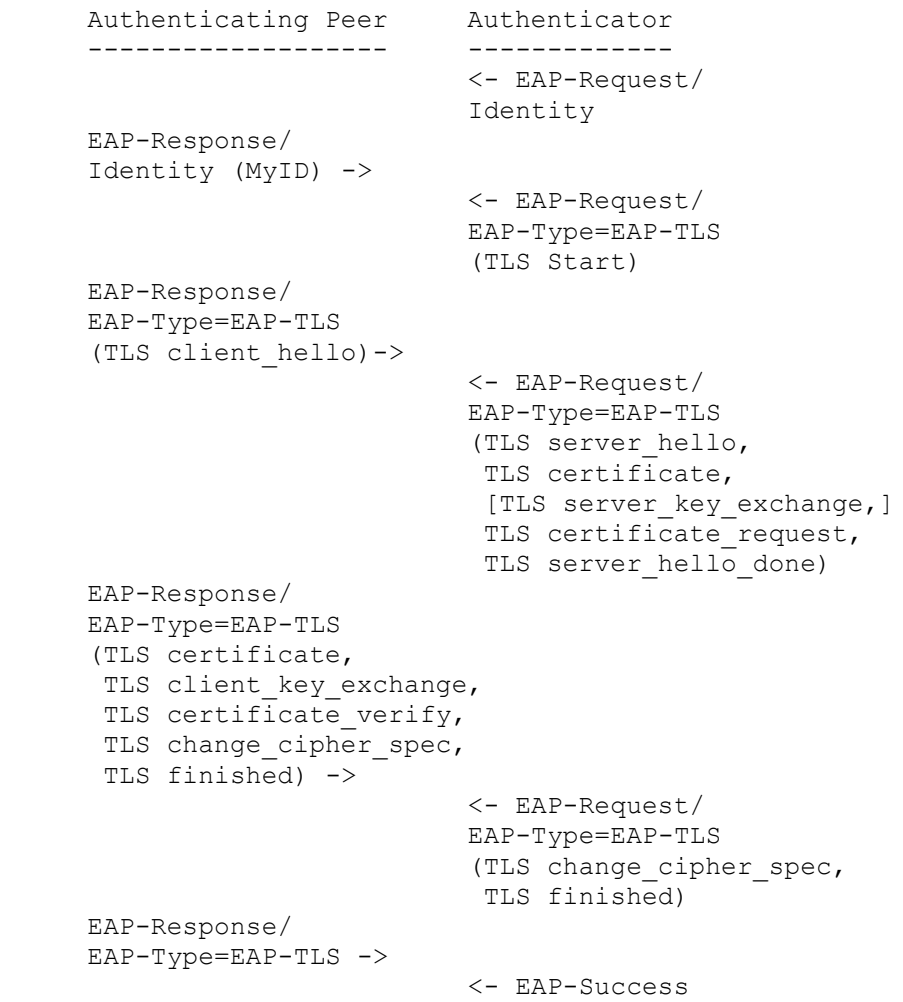


圖 3：EAP-TLS 交換

本文中的指南也適用於使用 EAP-TLS 進行網路訪問身份驗證。使用基於 TLS 的網路訪問認證解決方案的物聯網設備可以重用代碼的大部分內容，以便在應用層使用 DTLS / TLS，從而節省大量的閃存。但請注意，用於網路訪問身份驗證的憑據和用於應用程序層安全性的憑據很可能不同。

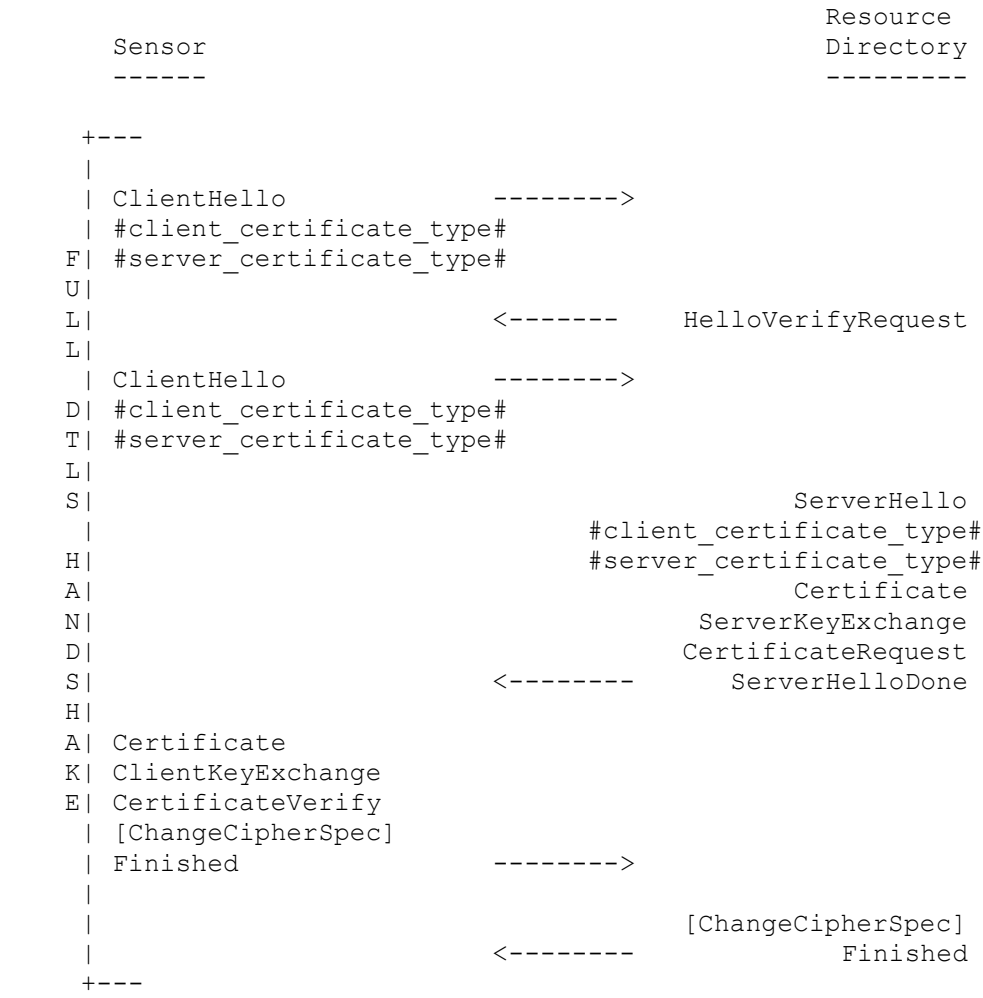
3.2.1.1.2. 基於 CoAP 的數據交換示例

當受約束的客戶端將傳感器數據上載到服務器基礎結構時，它可以通過 POST 消息將數據推送到服務器上的預配置端點來使用 CoAP。 在某些情況下，這可能過於局限，需要額外的功能，如圖 4 和圖 5 所示，其中 IoT 設備本身運行託管其他實體可訪問的資源的 CoAP 服務器。 儘管在物聯網設備上運行 CoAP 服務器，但它仍然是物聯網設備上的 DTLS 客戶端，它在我們的方案中啟動與非約束資源服務器的交互。

充當 DTLS 客戶端的傳感器與充當 DTLS 服務器的資源目錄間的初始 DTLS 交互將是完整的 DTLS 握手。握手完成後，雙方均已建立 DTLS 記錄層。隨後，CoAP 客戶端可以在資源目錄中安全註冊。

圖 4 顯示了一個傳感器與資源目錄開始 DTLS 交換，並使用 CoAP 註冊圖 5 中的可用資源。[CoRE-RD]將資源目錄（RD）定義為儲存有關 Web 資源的訊息並實現 Representational State 的 Web 實體傳輸（REST）介面，用於註冊和查找這些資源。 請注意，所描述的交換是從開放行動聯盟（OMA）Lightweight Machine-to-Machine（LWM2M）規範[LWM2M]借用的，該規範使用 RD 但增加了代理功能。

一段時間後（假設客戶端定期刷新其註冊），資源目錄接收來自應用程序的請求以從傳感器檢索溫度訊息。使用 GET 消息交換，該請求由資源目錄中繼到傳感器。已建立的 DTLS 記錄層可用於保護消息交換。



注意：標有“#”的擴充名隨 RFC 7250 一起引入。

圖 4：使用資源目錄的 DTLS / CoAP Exchange：
第 1 部分 - DTLS 交握

圖 5 顯示了使用 CoAP 在傳感器和資源目錄之間進行 DTLS 保護的通信。

圖 5：使用資源目錄的 DTLS / CoAP 交換：
第 2 部分 - CoAP / RD 交換

注意，使用先前建立的 DTLS 記錄層來保護從資源服務器發送的 CoAP GET 消息。

3.2.2. 受限制的 TLS / DTLS 服務器

第 3.2.1 節說明了部署模型，其中 TLS / DTLS 客戶端受到限制，需要採取措施來提高內存利用率，頻寬消耗，降低性能影響等。在本節中，我們假設受約束設備運行 TLS / DTLS 服務器，用於保護對在 CoAP，HTTP 或其他協定之上運行的應用程序層服務的訪問。 圖 6 說明了一種可能的部署，其中許多受約束的服務器正在等待常規客戶端訪問其資源。 整個過程很可能（但不一定）由第三方（身份驗證和授權服務器）控制。 此身份驗證和授權服務

器負責保存授權策略，該策略管理對資源的訪問和密鑰材料的散佈。

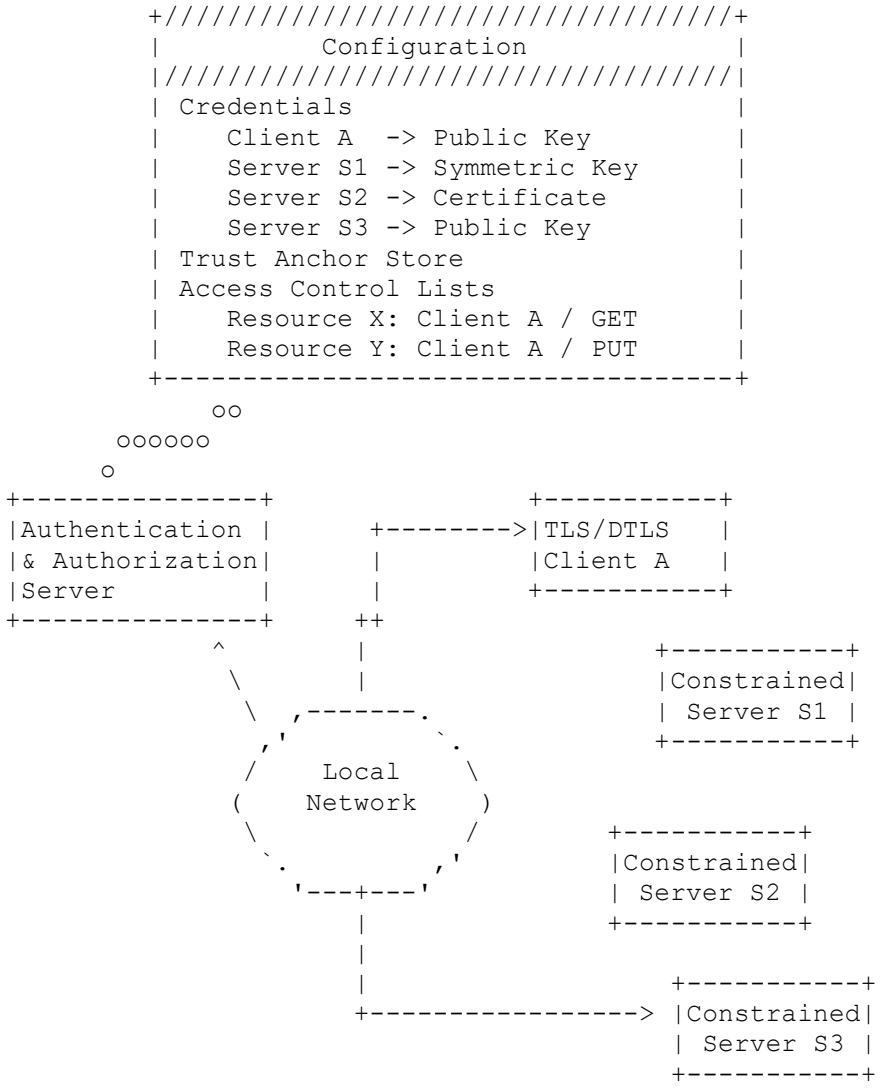


圖 6：約束服務器配置文件

具有受限服務器的部署必須克服多個挑戰。下面我們將解釋如何通過 CoAP 解決這些挑戰，作為一個例子。 其他協定可能提供類似的功能。 雖然在受限服務器上運行時，TLS / DTLS 協定配置文件的要求僅略有變化（與在受限客戶端上運行它相比），但其他一些生態系統因素將影響部署。

有幾個挑戰需要解決：

發現和可達性：

在啟動與服務器的連接之前，客戶端必須首先發現服務器。一旦發現，就需要維護設備的可達性。

在 CoAP 中，服務器提供的資源的發現是通過向眾所周知的 URI 發送單播或多播 CoAP GET 來完成的。 約束 RESTful 環境（CoRE）鏈接格式規範[RFC6690]描述了用例（參見第 1.2.1 節）並保留了 URI（參見第 7.1 節）。 CoAP 規範[RFC7252]的第 7 節描述了發現過程。 [RFC7390]描述了使用多播發現 CoAP 服務器的用例（參見第 3.3 節），並指定了 CoAP 組通信的協定處理規則（參見第 2.7 節）。

使用 RD [CoRE-RD]是發現已註冊服務器及其資源的另一種可能性。由於 RD 通常不是代理，因此客戶端可以發現向 RD 註冊的鏈接，然後直接訪問它們。

驗證：

下一個挑戰涉及為客戶端和服務器提供身份驗證憑據。 在 3.2.1 節中，我們假設將憑證（和其他配置訊息）提供給設備，並且這些憑證可以與授權服務器一起使用。 當然，這會導致客戶端與其服務器端基礎架構之間存在非常靜態的關係，但從部署的角度來看，這會帶來更少的挑戰，如[RFC7452]第 2 節所述。 無論如何，工程師和產品設計師必須確定如何將相關憑證散佈給各方。 例如，可能需要將共享機密提供給客戶端和受約束的服務器，以便隨後使用 TLS / DTLS PSK。在其他部署中，可能需要使用與基於憑證的身份驗證一起使用的憑證，私鑰和信任鏈。

實用的解決方案使用配對（也稱為印記）或受信任的第三方。通過配對，兩個設備執行未經身份驗證的特殊協定交換以建立共享密鑰（例如，使用未經驗證的 Diffie-Hellman（DH）交換）。 為了避免中間人攻擊，使用帶外信道來驗證沒有人篡改了交換的協定消息。 這個帶外頻道可以有多種形式，包括：

＊ 通過比較散列鍵，輸入密鑰和掃描 QR 碼來實現人為參與

* 使用替代無線通信信道（例如，除 Wi-Fi 之外的紅外線通信）

* 鄰近訊息

有關這些不同配對/壓印技術的更多詳細訊息，請參閱智慧對象安全研討會報告[RFC7397]以及就該主題提交的各種立場文件，例如[ImprintingSurvey]。受信任的第三方的使用遵循不同的方法，並且受限於圍繞環境的認證和授權（ACE）工作組[ACE-WG]中的持續標準化工作。

授權

最後一個挑戰是受限服務器在客戶端訪問受保護資源時做出授權決策的能力。 向受約束服務器預先提供訪問控制訊息可以是一種選擇，但僅適用於小規模，動態性較差的環境。 對於更細粒度和更動態的訪問控制解決方案，讀者可以參考 IETF ACE 工作組中正在進行的工作。

圖 7 顯示了一個示例交互，通過該交互，設備（在我們的情況下為恆溫器）在本地網路中搜索可發現的資源並訪問這些資源。 溫控器通過使用符合 RFC 6690 [RFC6690]的鏈接格式，使用“所有 CoAP 節點”多播地址使用鏈接本地發現消息來啟動該過程。 用於 CoAP 本地鏈接發現的 IPv6 組播地址是 FF02::FD。 結果，溫度傳感器和風扇回應。

這些回應允許恆溫器隨後通過發出給先前了解的端點的 CoAP GET 請求從溫度傳感器讀取溫度訊息。 在此示例中，我們假設訪問溫度傳感器讀數並控制風扇需要對恆溫器進行身份驗證和授權，並且 TLS 用於驗證兩個端點並確保通信安全。

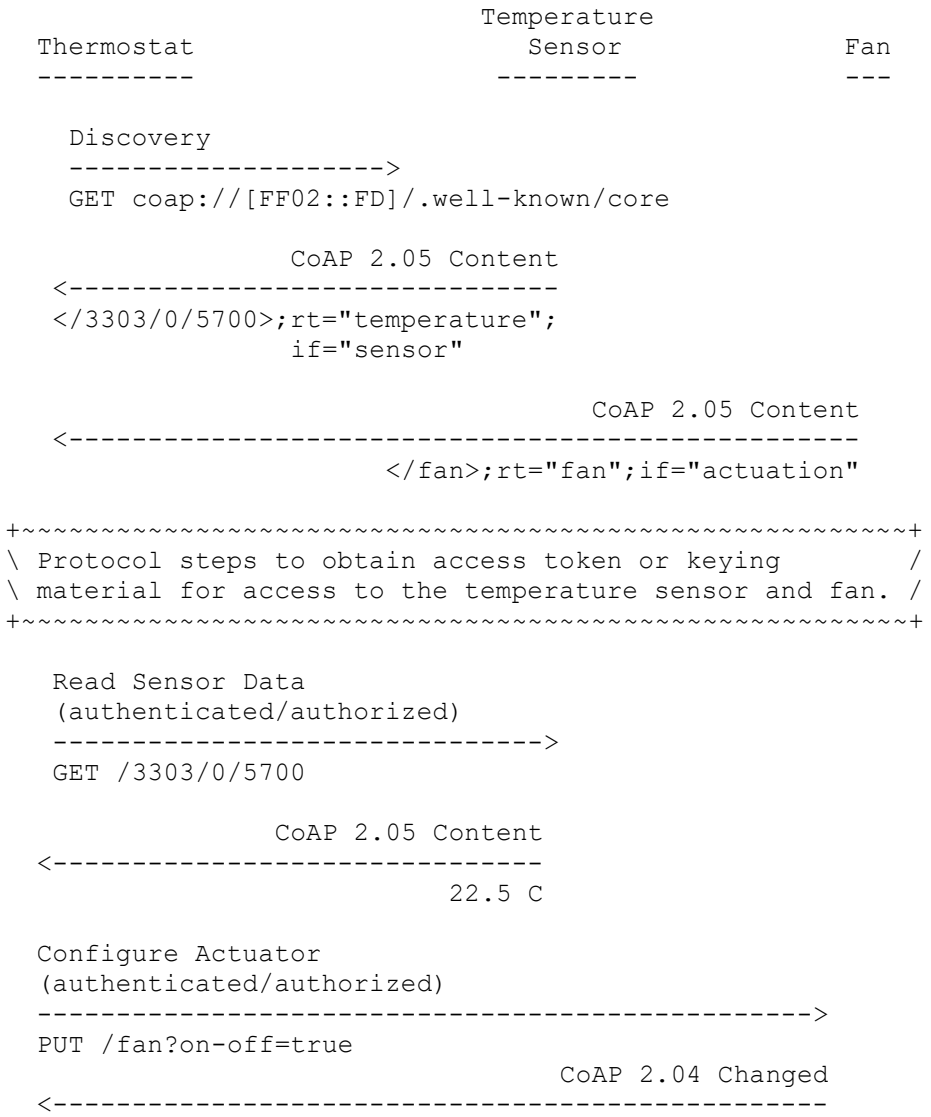


圖 7：本地發現和資源訪問

3.3. 密碼套件概念

TLS（以及 DTLS）支援密碼套件，並創建了 IANA 註冊表[IANA-TLS]來註冊套件。 密碼套件（以及定義它的規範）包含以下訊息：

- 驗證和密鑰交換算法（例如，PSK）
- 密碼和密鑰長度(例如,具有 128 位密鑰的高級加密標準(AES)[AES])

- 操作模式（例如，AES 的 CBC-MAC（CCM）模式的計數器）[RFC3610]

- 用於完整性保護的散列演算法，例如保全散列演算法（SHA）與密鑰散列訊息認證（HMAC）的組合（參見[RFC2104]和[RFC6234]）

- 用於偽隨機函數的散列算法（例如，帶有 SHA-256 的 HMAC）

- 雜項訊息（例如，認證標籤的長度）

- 有關密碼組件是適用於 DTLS 還是僅適用於 TLS 的訊息

例如，TLS 密碼套件 TLS_PSK_WITH_AES_128_CCM_8 使用預共享認證和密鑰交換算法。 [RFC6655]定義了這個密碼套件。 它使用 AES 加密算法，這是一種分組密碼。 由於 AES 算法支援不同的密鑰長度（例如 128,192 和 256 位），因此必須指定此訊息，並且所選密碼套件支援 128 位密鑰。 分組密碼以固定大小的塊加密明文，AES 以 128 位的塊大小運行。 對於超過 128 位的消息，消息被分區為 128 位塊，並且 AES 密碼通過適當的鏈接應用於這些輸入塊，這稱為操作模式。

TLS 1.2 引入了帶關聯數據的認證加密（AEAD）密碼套件（參見[RFC5116]和[RFC6655]）。 AEAD 是一類分組密碼模式，用於加密（部分）消息並同時驗證消息。 AES-CCM [RFC6655]是這種模式的一個例子。

某些 AEAD 密碼套件具有較短的身份驗證標籤（即消息身份驗證代碼），因此更適合於消息大小較小的低頻寬網路。以“_8”結尾的 TLS_PSK_WITH_AES_128_CCM_8 密碼套件具有 8 字節的身份驗證標籤，而常規 CCM 在撰寫本文時，密碼套件具有 16 個八位元組的身份驗證標籤。 CCM 的設計和安全屬性在[CCM]中進行了描述。

TLS 1.2 還替換了早期版本的 TLS 中使用的 TLS 偽隨機函數（PRF）中的 MD5 / SHA-1 散列函數與密碼組指定的 PRF 的組合。 出於

這個原因，最近的 TLS 1.2 密碼組規範的作者明確指出了 MAC 算法和與 TLS PRF 一起使用的散列函數。

4. 憑證類型

強制實施功能將取決於 IoT 設備使用的憑據類型。下面的小節描述了三種不同的憑據類型的含義，即預共享的機密，原始公共密鑰和證書。

4.1. 先決條件

後續部分中描述的所有交換都假定在 TLS / DTLS 交互開始之前已經散佈了一些訊息。憑證用於向服務器驗證客戶端,反之亦然。必須散佈哪些訊息項取決於所選的憑證類型。在所有情況下，物聯網設備需要知道要優先選擇哪種算法，特別是如果有多個算法選擇可用作實施密碼套件的一部分，以及有關其他通信端點的訊息（例如，以 URI 的形式）必須使用特定憑證。

預共享秘密：在這種情況下，需要將共享秘密與識別符一起提供給設備以及另一個通信方。

原始公鑰：公鑰和私鑰一起儲存在設備上，並且通常與某個識別符相關聯。 要驗證其他通信方，必須知道相應的憑證。 如果另一端也使用原始公鑰，則需要將其公鑰（帶外）配置到設備。

憑證：憑證的使用要求設備儲存公鑰（作為憑證的一部分）以及私鑰。 憑證將包含設備的識別符以及各種其他屬性。 假定兩個通信方都擁有包含 CA 憑證的信任鏈儲存，並且在憑證固定的情況下，擁有終端實體憑證。 與其他憑證類似，IoT 設備需要有關哪個實體使用哪個憑證的訊息。如果物聯網設備上沒有信任鏈儲存，則無法執行憑證驗證。

我們將上面列出的訊息稱為“設備憑證”，並且這些設備憑證可以在製造時間期間或之後的流程中提供給設備，這取決於預想的業務和部署模型。這些初始憑證通常稱為“信任根”。無論選擇哪種過程來生成這些初始設備憑證，都必須確保為每個設備配置

不同的密鑰對，並以盡可能安全的方式安裝。例如，最好在 IoT 設備本身上生成公鑰/私鑰，而不是在設備外部生成它們。由於 IoT 設備可能與各種其他方交互，因此初始設備憑證可能僅與某些專用實體一起使用，並且為設備配置其他配置和憑證留給單獨的交互。用於散佈憑證，訪問控制列表和配置訊息的專用協定的示例是 LWM2M 協定[LWM2M]。

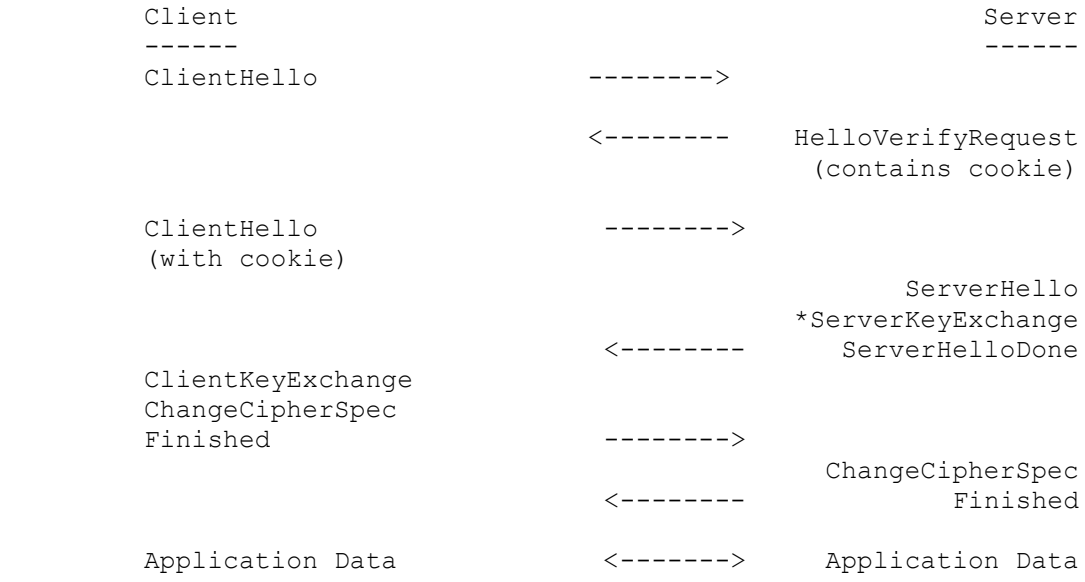
對於上面列出的所有憑據，可能需要替換或刪除這些憑據。雖然已經開發了單獨的協定來檢查這些憑證的狀態並管理這些憑證，例如信任鏈管理協定（TAMP）[RFC5934]，但是到目前為止，它們的使用並未在 IoT 上下文中設想。假設物聯網設備具有內置的軟體更新機制，它將允許更新低級設備訊息，包括憑證和配置參數。 但是，本文並未強制要求特定的軟體/韌體更新協定。

使用所有憑據作為 TLS / DTLS 身份驗證的輸入時，務必小心生成這些憑據。當使用預共享秘密時，關鍵考量是在密鑰生成期間使用足夠的熵，如[RFC4086]中所述。 從密碼，某些設備識別符或其他低熵源獲取共享密鑰並不安全。低熵秘密或密碼受字典攻擊。 在生成公鑰/私鑰對時也必須注意，因為缺乏隨機性會導致在許多設備中使用相同的密鑰對。 第 12 節也討論了該主題，因為密鑰也是在 TLS / DTLS 交換過程中生成的，並且適用相同的注意事項。

4.2. 預共享秘密

預共享秘密的使用是 TLS / DTLS 的最基本技術之一，因為它既具有計算效率又節省頻寬。RFC 4279 [RFC4279]中的 TLS 引入了基於預共享機密的身份驗證。

圖 8 說明了 DTLS 交換，包括 cookie 交換。雖然服務器不需要在每次交握時啟動 cookie 交換，但客戶端需要實現並在受到質詢時對其做出反應，如 RFC 6347 [RFC6347]中所定義。 cookie 交換允許服務器對洪水攻擊作出反應。



備註:

* 表示可選的消息有效負載

圖 8：包含 Cookie 交換的 DTLS PSK 身份驗證

請注意，[RFC4279]使用術語“PSK 標識”來指代用於指代適當秘密的識別符。雖然“識別符”在這種情況下更合適，但我們重用 RFC 4279 中定義的術語以避免混淆。 RFC 4279 不要求使用任何特定類型的 PSK 標識，並且客戶端和服務器必須就要使用的標識和密鑰達成一致。 RFC 4279 第 5.1 節中描述的身份的 UTF-8 編碼旨在改善人類使用 Web 瀏覽器提供的某些管理界面配置身份的情況的互操作性。但是，許多物聯網設備沒有用戶界面，並且大多數憑據都綁定到設備而不是用戶。此外，憑證通常被提供到硬體模塊中或與韌體一起提供。因此，編碼注意事項不適用於此使用環境。為了與此配置文件一起使用，PSK 標識不應採用結構化格式（例如域名，可分辨名稱或 IP 地址），並且服務器必須使用逐字節比較操作來執行與此相關的任何操作。 PSK 身份。根據 RFC 6943 [RFC6943]，這些類型的識別符稱為“絕對”。

在協定方面，客戶端通過在 ClientKeyExchange 消息中包括“PSK 標識”來指示它使用哪個密鑰。 如 3.2 節所述，客戶端可能具有多個預共享密鑰和單個服務器，例如，在託管上下文中。 TLS 服務器名稱指示（SNI）擴充允許客戶端傳達它正在聯繫的服務器

的名稱。 服務器實現需要基於從客戶端接收的 SNI 值來指導選擇。

RFC 4279 要求 TLS 實現支援 PSK 密碼套件，以支援最長 128 個八位元組的任意 PSK 身份和最長 64 個八位元組的任意 PSK 身份。對於在桌面和行動環境中使用的 TLS 堆疊，這是一個有用的假設，在這些環境中，管理介面用於提供身份和密鑰。 符合該配置文件的實現可以使用最長 128 個八位元組的 PSK 身份，以及最長 64 個八位元組的任意 PSK 身份。 建議使用較短的 PSK 身份。

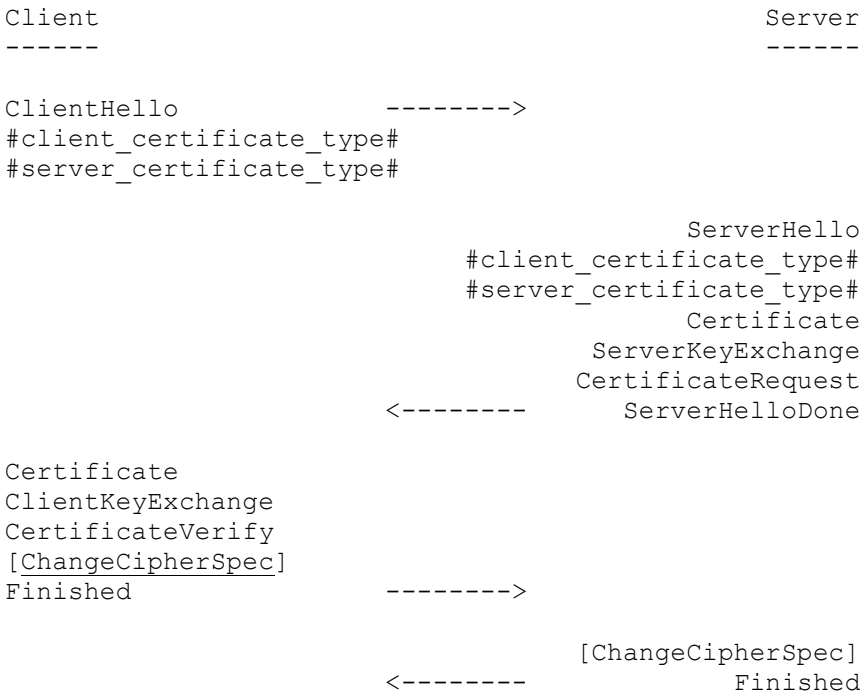
“受限制的應用協定（CoAP）” [RFC7252] 當前指定 TLS_PSK_WITH_AES_128_CCM_8 作為用於共享秘密的強制實施密碼套件。 該密碼套件使用 AES 算法，128 位密鑰和 CCM 作為操作模式。 標籤“_8”表示使用 8 個八位字節的認證標籤。 注意，短路認證標籤增加了不知道秘密密鑰的對手可以呈現具有將通過驗證過程的 MAC 的消息的機會。 接受偽造數據的可能性在[SP800-107-rev1]的第 5.3.5 節中進行了解釋，並且取決於使用給定密鑰的認證標籤的長度和允許的 MAC 驗證數量。

此密碼套件使用默認的 TLS 1.2 PRF，它使用具有 SHA-256 哈希函數的 HMAC。 注意：從 TLS 1.2（以及 DTLS 1.2）開始，密碼套件必須指定 PRF。 RFC 5246 聲明“新的密碼套件必須明確指定 PRF，並且通常應該使用帶有 SHA-256 的 TLS PRF 或更強的標準散列函數。” 本文中推薦的密碼套件使用 RFC 5246 第 5 節中定義的 SHA-256 構造。

符合本節中的配置文件的設備必須實現 TLS_PSK_WITH_AES_128_CCM_8 並遵循本節的指導。

4.3. 原始公鑰

如[RFC7250]中所定義的，使用帶有 TLS / DTLS 的原始公鑰是公鑰加密的第一個入口點，無需支付憑證和公鑰基礎結構（PKI）的價格。 規範重用現有的 Certificate 消息來傳達 SubjectPublicKeyInfo 結構中編碼的原始公鑰。為了表示支援，已經定義了兩個新擴充，如圖 9 所示，即 server_certificate_type 和 client_certificate_type。



註解：標有“#”的擴充名隨 RFC 7250 一起引入。

圖 9：DTLS 原始公鑰交換

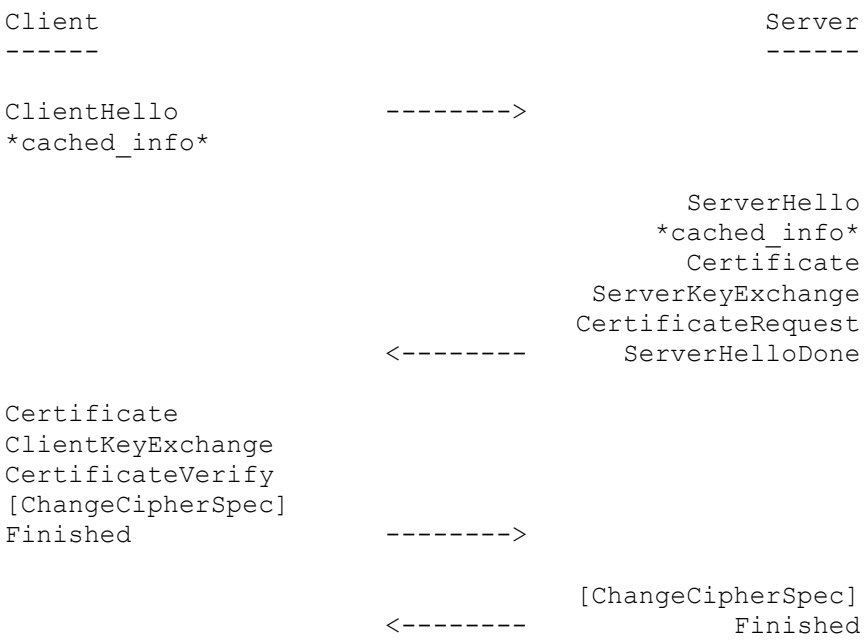
與此憑證類型一起使用的 CoAP 推薦密碼套件是 TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8 [RFC7251]。 這種基於橢圓曲線加密（ECC）的 AES-CCM TLS 密碼組使用短暫橢圓曲線 Diffie-Hellman（ECDHE）作為密鑰建立機制，使用橢圓曲線數位簽章算法（ECDSA）進行認證。 命名的 DH 組[FFDHE-TLS]不適用於此配置文件，因為它依賴於基於 ECC 的對應組件。 該密碼套件利用 DTLS 1.2 中的 AEAD 功能，並使用八位字節的認證標籤。 DH 密鑰交換的使用提供了完美的前向保密（PFS）。 有關 PFS 的更多詳細訊息，請參見第 9 節。

[RFC6090]為實現 ECC 算法提供了有價值的訊息，特別是用於選擇文獻中長時間（即 20 年或更長時間）可用的方法。

符合本節中的配置文件的設備必須實現 TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8 並遵循本節的指導。

4.4. 憑證

使用基於憑證的相互認證如圖 10 所示，它使用了"cached_info"擴充[RFC7924]。 需要支援"cached_info"擴充名。 緩存憑證鏈允許客戶端顯著減少通信開銷，否則服務器將為每個完整的 DTLS 交握提供最終實體憑證和憑證鏈。



註解：標有 “*”的擴充名隨 RFC 7924 一起引入。

圖 10：DTLS 基於憑證的相互認證

選擇基於 ECC 的密碼套件時，TLS / DTLS 提供了很多選擇。 本文限制了對 RFC 4492 [RFC4492]中定義的命名曲線的使用。在撰寫本文時，建議的曲線是 secp256r1，未壓縮點的使用遵循 CoAP 中的建議。 請注意，Curve25519（針對 ECDHE）的標準化正在進行中（參見[RFC7748]），將來可能需要對此曲線的支援。

符合本節中的配置文件的設備必須實現 TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8 並遵循本節的指導。

4.4.1. 服務器使用的憑證

RFC 6125 [RFC6125]中描述的用於驗證服務標識的算法對於確保在使用憑證時的適當安全性至關重要。 總之，該算法包含以下步驟：

1. 客戶端根據源域以及（可選）客戶端連接的服務類型構造可接受的引用識別符列表。

2. 服務器以 PKIX 憑證的形式提供其識別符。

3. 客戶端將其每個參考識別符都與提供的識別符進行比較，以查找匹配項。

4. 當對照提供的識別符檢查參考識別符時，客戶端將匹配識別符的源域以及（可選）它們的應用程序服務類型。

對於上面所示算法中使用的各種術語，請參閱 RFC 6125.重要的是要強調，需要將參考識別符與從憑證獲得的呈現識別符進行比較，以確保客戶端與目標服務器通信。

值得注意的是，算法描述和 RFC 6125 中的文字均假定使用了完全合格的 DNS 域名。 如果為服務器節點配置了完全限定的 DNS 域，則服務器證書必須包含完全限定的 DNS 域名或“ FQDN”作為 dNSName [RFC5280]。 對於 CoAP，[RFC7252]的 6.2 節中介紹了 coaps URI 方案。 如[RFC7252]的 9.1.3.3 節所述，此 FQDN 存儲在 SubjectAltName 或主題名稱的最左側公用名稱（CN）組件中，並由客戶端使用，以使其與查找過程中使用的 FQDN 匹配，如[RFC6125]中所述。 對於其他協定，必須參考適當的 URI 方案規範。

提供以下建議：

1. 憑證不得在憑證 CN 中使用 DNS 域名，而是使用 subjectAltName 屬性，如上一段所述。

2. 憑證不得包含帶通配符的域名。

3. 憑證不得包含多個名稱（例如，多個 dNSName 欄位）。

請注意，將存在未配置用於 DNS 域名的服務器，例如，為局域網中的附近設備提供資源的物聯網設備，如圖 7 所示。當使用此類受約束的服務器時，則使用 第 4.4.2 節中描述的憑證適用。 請注意，在這種情況下不能使用 SNI 擴充，因為 SNI 不提供傳送 64 位擴充唯一識別符（EUI-64）[EUI64]的能力。 請注意，本文不建議在憑證中使用 IP 地址，也不討論將 IP 地址放在憑證中的含義。

4.4.2. 客戶使用的憑證

對於客戶端憑證，SubjectAltName 中使用的識別符或主題名稱中最左側的 CN 組件必須是 EUI-64。

4.4.3. 憑證撤銷

對於憑證撤銷，既不使用在線憑證狀態協定（OCSP），也不使用憑證撤銷列表（CRL）。 相反，此配置文件依賴於軟體更新機制來提供有關已撤銷憑證的訊息。 雖然最近引入了多個 OCSP 裝訂[RFC6961]作為在 DTLS / TLS 交握內搭載 OCSP 請求/響應的機制（以避免單獨協定交握的成本），但還需要進一步調查以確定其對物聯網環境的適用性。

如本節前面所述，對信任鏈儲存的修改也取決於軟體更新機制。使用軟體更新機制存在限制，因為可能無法更改某些類型的密鑰，例如在製造期間提供的密鑰。 出於這個原因，製造商提供的憑證通常僅用於獲得進一步的憑證（例如，通過密鑰散佈服務器）以與 IoT 設備最終與之通信的服務器一起使用。

4.4.4. 憑證內容

表 1 中列出的所有憑證元素必須由聲稱支援基於憑證的身份驗證的客戶端和服務器實現。 本規範未使用其他憑證元素。

使用憑證時，IoT 設備必須支援至少為 3 的服務器憑證鏈，不包括信任鏈，並且可以拒絕提供超過 3 個鏈的服務器的連接。物聯網設備可以具有任意長度的客戶端憑證鏈。 顯然，較長的鏈需要

更多的數位簽章驗證操作來執行，並在 TLS 交握中導致更大的憑證消息。

表 1 提供了與此配置文件一起使用的憑證中元素的摘要。

Element	Notes
version	This profile uses X.509 v3 certificates [RFC5280].
serialNumber	Positive integer unique per certificate.
signature	This field contains the signature algorithm, and this profile uses ecdsa-with-SHA256 or stronger [RFC5758].
issuer	Contains the DN of the issuing CA.
validity	Values expressed as UTC time in notBefore and notAfter fields. No validity period mandated.
subject	See rules outlined in this section.
subjectPublicKeyInfo	The SubjectPublicKeyInfo structure indicates the algorithm and any associated parameters for the ECC public key. This profile uses the id-ecPublicKey algorithm identifier for ECDSA signature keys, as defined and specified in [RFC5480].
signatureAlgorithm	The ECDSA signature algorithm with ecdsa-with-SHA256 or stronger.
signatureValue	Bit string containing the digital signature.
Extension: subjectAltName	See rules outlined in this section.
Extension: BasicConstraints	Indicates whether the subject of the certificate is a CA and the maximum depth of valid certification paths that include this certificate. This extension is used for CA certs only, and then the value of the "cA" field is set to TRUE. The default is FALSE.
Extension: Key Usage	The KeyUsage field MAY have the following values in the context of this profile: digitalSignature or keyAgreement, keyCertSign for verifying signatures on public key certificates.
Extension: Extended Key Usage	The ExtKeyUsageSyntax field MAY have the following values in context of this profile: id-kp-serverAuth for server authentication, id-kp-clientAuth for client authentication, id-kp-codeSigning for code signing (for software update mechanism), and id-kp-OCSPSigning for future OCSP usage in TLS.

表 1：憑證內容

有各種加密演算法可用於簽署數字憑證; 這些演算法包括 RSA，數位簽章演算法（DSA）和 ECDSA。 如表 1 所示，在此配置文件中使用 ECDSA 對憑證進行簽名。這不僅適用於最終實體憑證，也適用於鏈中的所有其他憑證，包括 CA 憑證。 由於選項數量較少，此分析減少了 IoT 設備上儲存多個算法實現代碼所需的閃存量。

有關 X.509 憑證的更多詳細訊息，請參見[RFC7252]的第 9.1.3.3 節。

4.4.5. 客戶端憑證 URLs

RFC 6066 [RFC6066]允許避免發送客戶端憑證，而是使用 URL。這減少了無線傳輸。 請注意，TLS“cached_info”擴充不提供緩存客戶端憑證的任何幫助。

對於使用客戶端憑證且服務器端不受約束的環境，TLS / DTLS 客戶端必須實現對客戶端憑證 URL 的支援。對於受約束的服務器，不推薦使用此功能，因為它強制服務器執行額外的協定交換，可能使用它甚至不支援的協定。 由於額外的工作負載，使用此擴充還會增加針對受限服務器的 DoS 攻擊的風險。

4.4.6. 可信 CA 指示

RFC 6066 [RFC6066]允許客戶端指示其支援的信任錨。 通過基於證書的身份驗證，DTLS 服務器在 DTLS 握手期間將其最終實體證書傳送給客戶端。由於服務器不一定知道客戶端存儲的信任錨，因此為了促進證書路徑的構建和驗證，它在證書有效負載中包含中間 CA 證書。

現今，在大多數物聯網部署中，物聯網設備（及其上運行的軟體）與服務器端基礎設施之間存在相當靜態的關係。 對於物聯網設備與固定的，預配置的服務器集交互的這些部署，不推薦使用此擴充。

如果客戶端與動態發現的 TLS / DTLS 服務器進行交互，例如，在第 3.2.2 節中描述的用例中，則建議使用此擴充。

5. 簽署演算法延伸

RFC 5246 [RFC5246] 的第 7.4.1.4.1 節中定義的“signature_algorithms”擴充允許客戶端向服務器指示可以在數位簽章中使用哪些簽名/散列算法對。 客戶端必須發送此擴充以選擇使用 SHA-256，否則如果沒有此擴充，RFC 5246 默認為 SHA-1 / ECDSA 用於 ECDH_ECDSA 和 ECDHE_ECDSA 密鑰交換算法。

“signature_algorithms”擴充不適用於 4.2 節中描述的基於 PSK 的密碼套件。

6. 錯誤處置

TLS / DTLS 使用警報協定來傳達錯誤，並指定一長串錯誤類型。但是，並非 TLS / DTLS 規範中定義的所有錯誤消息都適用於此配置文件。通常，有兩類錯誤（如 RFC 5246 的 7.2 節所定義），即致命錯誤和警告。級別為“嚴重”的警報消息會導致連接立即終止。如果可能的話，開發人員應嘗試制定策略以應對這些致命錯誤，例如重新握手或使用（通常是有限的）用戶界面通知用戶。應用程序可能會忽略警告，因為許多物聯網設備要么記錄錯誤的方式有限，要么根本無法記錄錯誤。在任何情況下，實施者都必須仔細評估錯誤的影響以及糾正情況的方法，因為難以（或不可能）及時完成將決策委派給用戶的常用方法。

標記為 RESERVED 的所有錯誤消息僅支援向後兼容安全套接字層（SSL），並且不得與此配置文件一起使用。 其中包括 decryption_failed_RESERVED， no_certificate_RESERVED 和 export_restriction_RESERVED。

許多錯誤消息必須僅用於基於憑證的密碼套件。 因此，以下錯誤消息不得與 PSK 和原始公鑰認證一起使用：

- bad_certificate,

- unsupported_certificate,
- certificate_revoked,
- certificate_expired,
- certificate_unknown,
- unknown_ca, and
- access_denied.

由於此配置文件不在 TLS 層使用壓縮，因此也不得使用 decompression_failure 錯誤消息。

RFC 4279 為 PSK 密碼套件引入了新的警報消息“unknown_psk_identity”。如 RFC 4279 的第 2 節所述，也可以使用 decrypt_error 錯誤消息。對於此配置文件，TLS 服務器必須返回 decrypt_error 錯誤消息而不是 unknown_psk_identity，因為存在兩種機制並提供相同的功能。

此外，支援此規範的設備和服務器不應出現以下錯誤，但實現必須準備好處理這些錯誤以處理不符合本文中的配置文件的服務器：

協定版本：雖然本文僅關注 TLS / DTLS 協定的一個版本，即版本 1.2，但在撰寫本文時，正在進行的 TLS / DTLS 1.3 工作正在進行中。

安全性不足：此錯誤消息表明服務器要求密碼更安全。本文僅為每個配置文件指定一個密碼組，但隨著時間的推移，可能會增加額外的密碼組。

用戶取消：許多物聯網設備無人值守，因此不太可能出現此錯誤消息。

7. 會話恢復

會話恢復是核心 TLS / DTLS 規範的一項功能，允許客戶端繼續使用先前建立的會話狀態。 結果交換如圖 11 所示。此外，服務器可以選擇在恢復會話時不進行 cookie 交換。但是，客戶必須準備好與每次交握進行 cookie 交換。 cookie 交換未在圖中顯示。

圖 11：DTLS 對話恢復

受約束的客戶端必須實現會話恢復，以提高交握的性能。這將導致消息交換次數減少，計算開銷降低（因為在會話恢復交換期間僅使用對稱加密），並且會話恢復需要較少的頻寬。

對於服務器受限（但不是客戶端）的情況，客戶端必須實現 RFC 5077 [RFC5077]。 請注意，受約束服務器是指在 RAM 和閃存方面存在限制的設備，這會限制可以儲存在此類設備上的 TLS / DTLS 安全狀態訊息的數量。 RFC 5077 指定了 TLS / DTLS 會話恢復的版本，該版本不要求受約束服務器維護每個會話的狀態訊息。 這是通過使用基於票證的方法實現的。

如果客戶端和服務器都是受約束的設備，則兩個設備都應該實現 RFC 5077 並且必須實現基本的會話恢復。不想使用會話恢復的客戶端始終能夠發送帶有空 session_id 的 ClientHello 消息以恢復為完全交握。

8. 壓縮

[RFC7525]的第 3.3 節建議禁用由於攻擊引起的 TLS / DTLS 級壓縮，例如 CRIME [CRIME]。 對於物聯網應用，不需要在 TLS / DTLS 層進行壓縮，因為應用層協定是高度優化的，並且 DTLS 層的壓縮算法增加了代碼大小和復雜性。

此 TLS / DTLS 配置文件不推薦 TLS / DTLS 層壓縮。

9. 完美的前向保密

即使在長期機密受到威脅的情況下，PFS 仍可保留過去協定交互的機密性。

第 4.2 節中建議的 PSK 密碼套件不提供此屬性，因為它不利用 DH 交換。 支援 PFS 進行基於 PSK 的身份驗證的新密碼套件（如[PSK-AES-CCM-TLS]中提出的）可能會在不久的將來以標準化密碼套件的形式提供。 推薦的基於 PSK 的密碼套件具有出色的性能，非常小的內存佔用空間，並且在線路開銷方面最低，以不使用任何公共加密為代價。 對於可以接受公鑰密碼學的部署，原始公鑰的使用可能在帶外驗證的 PSK 密碼套件和非對稱密碼學提供的功能之間提供中間立場。

物理攻擊為獲取對儲存在 IoT 設備上的加密材料的訪問創造了額外的機會。PFS 密碼套件可防止攻擊者在成功進行長期密鑰洩露之前獲得交換的通信內容; 然而，（出於性能或能效原因）在多個不同會話中重複使用相同的短暫 DH 密鑰的實現部分地使 PFS 失效，從而增加了損壞程度。 出於這個原因，實現不應該在多個協定交換上重用短暫的 DH 密鑰。

在選擇不支援 PFS 屬性的密碼套件時，必須考慮披露過去的通信交互的影響和增加普遍監控成本的期望（如[RFC7258]所要求的）。

聲稱支援此配置文件的客戶端實現必須根據所選憑據類型實現第 4 節中列出的密碼套件。

10. 保活

應用層通信可以在端點處創建狀態，並且該狀態可能在某個時間到期。因此，如有必要，應用程序會定義刷新狀態的方法。雖然應用層交換在很大程度上超出了底層 TLS / DTLS 交換的範圍，但類似的狀態考量也在 TLS / DTLS 層面發揮作用。雖然 TLS / DTLS 還在客戶端和服務器上以安全上下文的形式創建狀態（請參閱 RFC 5246 中附錄 A.6 中描述的安全性參數），但此狀態訊

息不會過期。但是，網路中介也可能會分配狀態並要求此狀態保持活動狀態。如果無法在狀態包過濾防火牆或 NAT 處保持狀態，則可能導致一個節點無法到達另一個節點，因為這些中間盒會阻止資料封包。在 TLS / DTLS 客戶端和服務器之間交換的定期保持活動消息使這些中間盒的狀態保持活動狀態。根據 [HomeGateway]中描述的測量，住宅閘道器中使用的狀態管理實踐存在一些差異，但超時受到傳輸層協定選擇的嚴重影響：UDP 的超時通常比 TCP 的超時短得多。

RFC 6520 [RFC6520]定義了一種心跳機制，用於測試另一個對等體是否仍然存活。作為附加特徵，也可以使用相同的機制來執行路徑最大傳輸單元（MTU）發現。

關於使用 RFC 6520 的建議取決於 IoT 設備執行的消息交換類型以及應用程序作為其應用程序功能的一部分需要交換的消息數。有三種類型的交換需要分析：

客戶端啟動的一次性消息

這是一種常見的通信模式，其中 IoT 設備不定期地將數據上載到網際網路上的服務器。 通信可以由特定事件觸發，例如打開門。

在 IP 地址改變的情況下，可能需要重新啟動 DTLS 交握（理想情況下，如果可能，使用會話恢復）。

在這種情況下，此方案沒有使用保持活動擴充。

客戶端啟動的常規數據上傳

這是前一種情況的變體，其中數據定期上傳，例如，基於頻繁的溫度讀數。 如果 NAT 綁定和 IP 地址都沒有發生變化，那麼記錄層將不會發現任何變化。對於 IP 地址和端口號更改的情況，必須使用會話恢復重新創建記錄層。

在這種情況下，保持活動擴充沒有用處。 設備也很可能在數據傳輸之間進入睡眠週期以保持低功耗。

服務器啟動的消息

在前兩個場景中，客戶端啟動協定交互並對其進行維護。 由於到客戶端的消息可能被中間件阻止，因此初始連接設置由客戶端觸發，然後由服務器保持活動狀態。

對於此消息交換模式，使用 DTLS 心跳消息非常有用，但可能必須與應用程序交換協調（例如，當使用 CoAP 資源目錄時）以避免冗餘保持活動消息交換。 MTU 發現機制也是[RFC6520]的一部分，不太可能具有相關性，因為對於許多物聯網部署，最受約束的鏈路是物聯網設備和網路本身之間的無線介面（而不是一些鏈路） - 結束路徑）。 僅在更複雜的網路拓撲中，例如多跳網狀網路，路徑 MTU 發現可能是合適的。 還必須注意，DTLS 本身已經通過使用交握協定的分段功能提供了基本的路徑發現機制（參見 RFC 6347 的第 4.1.1.1 節）。

對於服務器啟動的消息，建議使用心跳擴充。

11. 逾時

有多種有線和無線技術可用於將設備連接到網際網路。許多低功率無線電技術，例如 IEEE 802.15.4 或藍牙智慧，僅支援小幀大小（例如，如[RFC4919]中所解釋的，在 IEEE 802.15.4 的情況下為 127 字節）。 其他無線電技術，例如使用短消息服務（SMS）的全球行動通信系統（GSM），在有效載荷大小方面具有類似的約束，例如 140 字節而沒有被稱為“連接 SMS”的可選分段和重組方案。 ，但顯示更高的延遲。

DTLS 交握協定為 TLS 交握協定添加了分段和重組機制，因為每個 DTLS 記錄必須適合單個傳輸層資料封包，如[RFC6347]的第 4.2.3 節所述。由於交握消息可能大於最大記錄大小，因此該機制在多個 DTLS 記錄上分割交握消息，每個 DTLS 記錄可以單獨發送。

為了處理 UDP 提供的不可靠的消息傳遞，DTLS 增加了超時和“每次飛行”重傳，如[RFC6347]的 4.2.4 節所述。儘管超時值是特定於實現的，但[RFC6347]的第 4.2.4.1 節中提供了建議，初始定時器值為 1 秒，每次重傳時的值加倍，最多不超過 60 秒。

由於受約束側的處理時間較長，TLS 協定步驟可能需要更長的時間。 另一方面，DTLS 處理重傳的方式（每個網段而不是每個網段）往往與低頻寬網路交互不良。

由於這些原因，必須使虛假重傳的概率最小化，並且在超時時，發送端點不會過於積極地做出反應。當無線傳感器網路（WSN）暫時擁塞時，後者尤為重要：如果丟失的資料封包重新注入太快，則擁塞會惡化。

因此建議初始計時器值為 9 秒，指數後退不小於 60 秒。

選擇該值足夠大以吸收由於受約束端點上的慢計算或固有網路特性（例如，GSM-SMS）而導致的大延遲變化，以及產生少量重傳事件並放鬆它們之間的起搏。 其最壞情況等待時間與使用 1s 超時（即 63s）相同，而觸發少於一半的重傳（2 而不是 5）。

為了最小化 DTLS 交握期間的喚醒時間，睡眠節點可能決定選擇較低的閾值，並因此選擇較小的初始超時值。如果是這種情況，實現必須考慮本節中描述的有關網路穩定性的考量。

12. 隨機數生成

TLS / DTLS 協定要求在協定運行期間提供隨機數。 例如，在 ClientHello 和 ServerHello 交換期間，客戶端和服務器交換隨機數。此外，DH 密鑰交換的使用在密鑰對生成期間需要隨機數。

重要的是要注意，許多物聯網設備上沒有可用於筆記本電腦或台式機上隨機池的資源，例如鼠標移動，擊鍵時間，硬盤驅動器頭行動時的空氣流等。其他來源必須找到或必須添加專用硬體。

缺乏嵌入式系統中的隨機性源可能導致一次又一次地生成相同的密鑰。

ClientHello 和 ServerHello 消息包含 “Random”結構，它有兩個組件：gmt_unix_time 和 28 個隨機字節的序列。gmt_unix_time 以標準 UNIX 32 位格式保存當前時間和日期（從 1970 年 1 月 1 日格林威治標準時間開始的午夜起的秒數）。由於許多物聯網設備無法訪問準確的時鐘，因此建議 ClientHello 或 ServerHello 的接收方不要假設 “Random.gmt_unix_time”中的值是當前時間的準確表示，而是將其視為不透明的隨機字符串。

當 TLS 與基於憑證的身份驗證一起使用時，需要時間訊息的可用性來檢查憑證的有效性。更高層協定可以提供安全的時間訊息。ServerHello 的 gmt_unix_time 組件不用於此目的。

使用 TLS / DTLS 的物聯網設備必須提供生成高質量隨機數的方法。將基於硬體的隨機數發生器集成到產品中有多種實現選擇：微控制器內部的實現是一種選擇，但專用的加密晶片也是合理的選擇。 最佳選擇取決於本文範圍之外的各種因素。 隨機數生成的準則和要求可以在 RFC 4086 [RFC4086]和 NIST 特刊 800-90a [SP800-90A]中找到。

我們強烈建議晶片製造商為隨機數生成器提供足夠的設計文件，以便客戶對生成的隨機數的質量有信心。通過提供有關用於驗證硬體模塊生成的數字隨機性的過程的訊息，可以增加可信度。例如，NIST Special Publication 800-22b [SP800-22b]描述了可用於驗證隨機數發生器的統計測試。

13. 截斷的 MAC 和 Encrypt-then-MAC 擴充

截斷的 MAC 擴充在 RFC 6066 [RFC6066]中引入，目標是減小記錄層使用的 MAC 的大小。 此擴充是為使用較舊操作模式的 TLS 密碼套件開發的，其中 MAC 和加密操作是獨立執行的。

本文中推薦的密碼套件使用較新的 AEAD 結構，即具有 8 個八位字節認證標籤的 CCM 模式，因此不適用於截斷的 MAC 擴充。

RFC 7366 [RFC7366]引入了 encrypt-then-MAC 擴充（而不是先前使用的 MAC-then-encrypt），因為 MAC-then-encrypt 機制已經成

為許多安全漏洞的主題。 但是，RFC 7366 也不適用於本文中推薦的 AEAD 密碼。

符合此規範的實現必須使用 AEAD 密碼。 RFC 7366（“encrypt-then-MAC”）和 RFC 6066（“截斷的 MAC 擴充”）不適用於本規範，不得使用。

14. 服務器名稱指示（SNI）

SNI 擴充[RFC6066]定義了一種機制，供客戶端告訴 TLS / DTLS 服務器它想要聯繫的服務器的名稱。 對於許多託管環境而言，這是一個有用的擴充，其中多個虛擬服務器在單個 IP 地址上運行。

除非已知 TLS / DTLS 客戶端不與託管環境中的服務器交互，否則必須實現服務器名稱指示擴充。

15. 最大片段長度協商

此 RFC 6066 擴充將記錄層所需的最大片段長度支援從 2 ^ 14 字節降低到 2 ^ 9 字節。

這是一個非常有用的擴充，允許客戶端向服務器指示它用於傳入消息的最大內存緩衝區數。 最終，此擴充的主要好處是允許客戶端實現降低其 RAM 要求，因為客戶端不需要接受大容量的資料封包（例如普通 TLS / DTLS 所要求的 16K 資料封包）。

客戶端實施必須支援此擴充。

16. 對話散列

為了開始連接保護，記錄協定需要指定一套算法，主密鑰以及客戶端和服務器隨機值。 用於計算主密鑰的算法在 RFC 5246 的第 8.1 節中定義，但它僅包括在交握期間交換的少量參數，並且不包括諸如客戶端和服務器標識之類的參數。這可以被攻擊者用來發起中間人攻擊，因為不能保證主密鑰在會話中是唯一的，如在“三次交握”攻擊[Triple-HS]中發現的那樣。

[RFC7627]定義了一個 TLS 擴充，它將主密鑰綁定到計算它的完整交握的日誌，從而防止此類攻擊。

客戶端實現應該實現此擴充，即使此配置文件推薦的密碼套件不容易受到此攻擊。 對於基於 DH 的密碼套件，密鑰材料由雙方提供，並且在預共享密鑰密碼套件的情況下，雙方需要擁有共享秘密以確保交握成功完成。 但是，某些應用層協定可能會在 DTLS 之上隧道傳輸其他身份驗證協定，從而使這種攻擊再次發生。

17. 重新談判攻擊

TLS / DTLS 允許已經具有 TLS / DTLS 連接的客戶端和服務器通過使用重新協商功能來協商新參數，生成新密鑰等。 重新協商發生在現有連接中，新的交握包與應用程序數據一起被加密。 完成重新協商程序後，新頻道將替換舊頻道。

如 RFC 5746 [RFC5746]中所述，兩次交握之間沒有加密綁定，儘管新交握是使用原始交握建立的加密參數執行的。

為防止重新協商攻擊 [RFC5746]，此規範要求禁用 TLS 重新協商功能。 客戶端必須通過警報消息（no_renegotiation）響應服務器啟動的重新協商嘗試，並且客戶端不得啟動它們。

18. 降級攻擊

當客戶端發送的 ClientHello 版本高於服務器已知的最高版本時，服務器應該使用等於服務器已知的最高版本的 ServerHello.version 進行回復，然後交握可以繼續。 此行為稱為版本容差。 版本不容忍是指服務器（或中間件）在看到 ClientHello.version 高於其所知的時間時中斷交握的時間。這是導致某些客戶端使用較低版本重新運行交握的行為。 因此，當系統運行舊的 TLS / SSL 版本時（例如，由於需要與遺留系統集成），引入了潛在的安全漏洞。 在最壞的情況下，這允許攻擊者將協定交握降級到 SSL 3.0。 SSL 3.0 非常破碎，因此沒有可用的安全密碼（參見[RFC7568]）。

通過 TLS 後降發信密碼套件值（SCSV）[RFC7507]擴充來解決上述降級漏洞。但是，該解決方案不適用於符合此配置文件的實現，因為版本協商必須使用 TLS / DTLS 版本 1.2（或更高版本）。 更具體地說，這意味著：

- 客戶端不得在 ClientHello 中發送低於 1.2 版的 TLS / DTLS 版本。

- 客戶端不得重試提供低於 1.2 的 TLS / DTLS 版本的失敗協商。

- 如果無法協商 TLS / DTLS 版本> = 1.2，服務器必須通過發送 protocol_version 致命警報來使交握失敗。 請注意，中止的連接是不可恢復的。

19. 加密彈性

本文建議軟體和晶片製造商實施 AES 和 CCM 操作模式。本文引用了 CoAP 推薦的密碼套件選擇，這些選擇是根據物聯網社群的實施和部署經驗選擇的。 但是，隨著時間的推移，對算法的偏好會發生變化。 並非密碼組的所有組件都可能以相同的速度發生變化。 密碼、操作模式和散列算法更可能發生變化。 建議的密鑰長度也必須隨時間調整。 某些部署環境也會受到本地監管的影響，這可能會規定某種算法和密鑰大小組合。 有關選擇特定 ECC 曲線的持續討論也可能會影響實施。 請注意，本文不建議或強制要求特定的 ECC 曲線。

向晶片製造商提出以下建議：

- 使任何基於 AES 硬體的加密實現可供在協定疊中更高層的安全實現上工作的開發人員訪問。 有時，硬體實現被添加到微控制器中，以提供對鏈路層所需功能的支援，並且僅適用於片上鍊路層協定實現。 這樣的設置不允許應用程序開發人員重用基於硬體的 AES 實現。

- 為加密功能的使用提供靈活性，並考慮到未來的可擴充性。例如，為開發人員提供 AES-CCM 實現是第一步，但由於

AES-CCM 實現之間的參數差異，這樣的實現可能無法使用。 IEEE 802.15.4 和 Bluetooth Smart 中的 AES-CCM 使用 13 個八位元組的隨機數長度，而 DTLS 使用 12 個八位元組的隨機數長度。 因此，用於 IEEE 802.15.4 和藍芽智慧的 AES-CCM 的硬體實現不能由 DTLS 堆疊重用。

- 除了（或作為替代）完整功能之外，還可以訪問構建塊。 例如，為開發人員提供 AES 加密功能的晶片製造商可以使用它來構建有效的 AES-GCM 實現。 另一個例子是提供一個特殊指令，以提高加速無進位乘法的速度。

作為開發人員和產品架構師的建議，我們建議提供足夠的空間，以便在產品的整個生命週期內升級到更新的加密算法。 例如，雖然在整個規範中推薦使用 AES-CCM，但未來的產品可能會將 ChaCha20 密碼與 Poly1305 驗證器[RFC7539]結合使用。 此假設建立在強大的軟體更新機制提供。

20. 關鍵長度建議

RFC 4492 [RFC4492]基於用於攻擊它們的最著名算法，給出了對稱密鑰和非對稱密鑰密碼系統的近似可比密鑰大小。 雖然其他出版物的數字略有不同，例如[Keylength]，但近似關係仍然適用。 圖 12 以位為單位說明了可比較的密鑰大小。

Symmetric	ECC	DH/DSA/RSA
80	163	1024
112	233	2048
128	283	3072
192	409	7680
256	571	15360

圖 12：基於 RFC 4492 的可比較密鑰大小（以 Bits 為單位）

在撰寫本文時，在[RFC7525]中發現的與基於 TLS 的密碼一起使用的密鑰大小建議建議至少 2048 位的 DH 密鑰長度，它對應於 112 位對稱密鑰和 233 位 ECC 密鑰。 這些建議與其他組織的建議大致相符，例如美國國家標準技術研究院（NIST）或歐洲網路和訊息安全局（ENISA）。 [ENISA-Report2013]的作者補充說，在未來幾年中，80 位對稱密鑰足以滿足傳統應用程序的需要，但是 128 位對稱密鑰是部署新系統的最低要求。 作者進一步指出，還

需要考慮到需要確保數據安全的時間長度。在不久的將來，可以將 80 位對稱密鑰用於事務性數據，而對於持久性數據則必須堅持使用 128 位對稱密鑰。

請注意，在 IoT 設備具有較長的預期生命週期（例如 10 年以上）的假設下，保守地選擇 112 位對稱密鑰的建議，並且該密鑰長度建議是指用於設備認證的長期密鑰。考慮到交換數據的靈敏度，為了保護事務數據（例如在各種 TLS / DTLS 密碼組中使用的短暫 DH 密鑰）而動態提供的密鑰可以更短。

21. 假啟動

[RFC5246]中指定的完整 TLS 交握在交握完成之前需要兩個完整的協定輪次（四個行程），並且協定方可以開始發送應用數據。

在三次行程後完成縮寫交握（恢復較早的 TLS 會話），因此如果客戶端首先發送應用程序數據，則僅添加一次往返時間。在三次行程後完成縮寫交握（恢復較早的 TLS 會話），因此如果客戶端首先發送應用程序數據，則僅添加一次往返時間。

如果滿足[TLS-FALSESTART]中列出的條件，則可以在發件人發送自己的“ChangeCipherSpec”和“Finished”消息時傳輸應用程序數據。如果客戶端首先發送應用程序數據並且如果服務器首先發送應用程序數據則用於縮略交握，則這實現了完全交握的一個往返時間的改進。

本文中基於公鑰的密碼套件滿足使用 TLS False Start 假啟動機制的條件。 總之，條件是：

- 現代對稱密碼，有效密鑰長度為 128 位，例如 AES-128-CCM
- 客戶端憑證類型，例如 ecdsa_sign
- 密鑰交換方法，例如 ECDHE_ECDSA

基於對完整 TLS / DTLS 交換的完整往返的改進，此規范建議在客戶端首先發送應用程序數據時使用 False Start 假啟動機制。

22. 隱私考量

DTLS 交握交換傳達各種識別符，這些識別符可由路徑上的竊聽者觀察到。例如，DTLS PSK 交換顯示 PSK 標識、支援的擴充、會話 ID、算法參數等。當使用會話恢復時，可以通過路徑上的對手來關聯各個 TLS 會話。 由於在這些設備上更新軟體的成本，在許多物聯網部署中，密鑰材料及其識別符很可能在更長的時間內持久存在。

由於許多物聯網設備最初是由人提供的，因此許多物聯網設備在無人看管的情況下運行，因此用戶參與在許多物聯網部署中構成了挑戰。必須在系統級別而不是 DTLS 交換本身的級別提供控制數據共享和配置首選項的能力，這是本文的範圍。很自然地，將 DTLS 與相互身份驗證結合使用將允許 TLS 服務器收集有關 IoT 設備的身份驗證訊息（可能需要很長時間）。儘管這種強大的身份驗證形式可以防止歸因錯誤，但它也可以進行強大的身份驗證。與其他數據類型相關聯的與設備相關的數據收集（例如，傳感器記錄）將被證明是真正有用的，但是這些額外的數據可能包括有關設備所有者的個人訊息或有關其感測到的環境的數據。因此，存儲在服務器端的數據將很容易受到存儲數據洩露的影響。對於客戶端和服務器之間的通信，此規範可防止竊聽者訪問通信內容。雖然基於 PSK 的密碼套件不提供 PFS，但非對稱版本提供。當獲得對長期秘密的訪問時，這防止了對手獲得過去的通信內容。請注意，本文中的建議並未提供使流量分析更加困難的額外精力。

請注意，通信本身的缺失或存在可能會向對手洩露訊息。例如，存在傳感器可以在人進入建築物時發起消息傳遞。 雖然 TLS / DTLS 將為傳輸的訊息提供機密性保護，但它無助於隱藏所有通信模式。 此外，不受 TLS / DTLS 保護的 IP 標頭還顯示有關其他通信端點的訊息。 對於存在此類隱私問題的應用程序，需要額外的安全措施，例如注入虛擬流量和洋蔥路由。 對此類解決方案的詳細處理超出了本文的範圍，需要系統級觀點。

23. 安全考量

整個文件都是關於安全性的。

整個文件都是關於安全性的。我們還要指出，在物聯網系統中設計軟體更新機制對於確保可以增強功能和修復潛在漏洞至關重要。 此軟體更新機制對於更改配置訊息非常重要，例如，信任鏈和其他與鍵控相關的訊息。 OMA [LWM2M]發布的 LWM2M 協定提供了這種合適的軟體更新機制。

24. 參考文獻

24.1. 規範性參考文獻

[EUI64] IEEE, "Guidelines for 64-bit Global Identifier (EUI-64)", Registration Authority,<https://standards.ieee.org/regauth/oui/tutorials/EUI64.html>.

[GSM-SMS] ETSI, "3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Technical realization of the Short Message Service (SMS) (Release 13)", 3GPP TS 23.040 V13.1.0, March 2016.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119,DOI 10.17487/RFC2119, March 1997,<http://www.rfc-editor.org/info/rfc2119>.

[RFC4279] Eronen, P., Ed. and H. Tschofenig, Ed., "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)",RFC 4279, DOI 10.17487/RFC4279, December 2005,<http://www.rfc-editor.org/info/rfc4279>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246,DOI 10.17487/RFC5246, August 2008,<http://www.rfc-editor.org/info/rfc5246>.

[RFC5746] Rescorla, E., Ray, M., Dispensa, S., and N. Oskov, "Transport Layer Security (TLS) Renegotiation Indication Extension", RFC 5746,

DOI 10.17487/RFC5746, February 2010,<http://www.rfc-editor.org/info/rfc5746>.

[RFC6066] Eastlake 3rd, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066,DOI 10.17487/RFC6066, January 2011,<http://www.rfc-editor.org/info/rfc6066>.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <http://www.rfc-editor.org/info/rfc6125>.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347,January 2012, <http://www.rfc-editor.org/info/rfc6347>.

[RFC6520] Seggelmann, R., Tuexen, M., and M. Williams, "Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS) Heartbeat Extension", RFC 6520,DOI 10.17487/RFC6520, February 2012,<http://www.rfc-editor.org/info/rfc6520>.

[RFC7250] Wouters, P., Ed., Tschofenig, H., Ed., Gilmore, J., Weiler, S., and T. Kivinen, "Using Raw Public Keys in Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", RFC 7250, DOI 10.17487/RFC7250,June 2014, <http://www.rfc-editor.org/info/rfc7250>.

[RFC7251] McGrew, D., Bailey, D., Campagna, M., and R. Dugal, "AES- CCM Elliptic Curve Cryptography (ECC) Cipher Suites for TLS", RFC 7251, DOI 10.17487/RFC7251, June 2014,<http://www.rfc-editor.org/info/rfc7251>.

[RFC7627] Bhargavan, K., Ed., Delignat-Lavaud, A., Pironti, A., Langley, A., and M. Ray, "Transport Layer Security (TLS) Session Hash and Extended Master Secret Extension",RFC 7627, DOI 10.17487/RFC7627, September 2015,<http://www.rfc-editor.org/info/rfc7627>.

[RFC7924] Santesson, S. and H. Tschofenig, "Transport Layer Security (TLS) Cached Information Extension", RFC 7924,DOI 10.17487/RFC7924, July 2016,<http://www.rfc-editor.org/info/rfc7924>.

[WAP-WDP] Open Mobile Alliance, "Wireless Datagram Protocol", Wireless Application Protocol, WAP-259-WDP, June 2001.

24.2. 訊息參考

[ACE-WG] IETF, "Authentication and Authorization for Constrained Environments (ACE) Working Group",<https://datatracker.ietf.org/wg/ace/charter>.

[AES] National Institute of Standards and Technology, "Advanced Encryption Standard (AES)", NIST FIPS PUB 197, November 2001, <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>.

[CCM] National Institute of Standards and Technology, "Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality", NIST Special Publication 800-38C, May 2004,<http://csrc.nist.gov/publications/nistpubs/800-38C/SP800-38C_updated-July20_2007.pdf>.

[COAP-TCP-TLS]

Bormann, C., Lemay, S., Tschofenig, H., Hartke, K., Silverajan, B., and B. Raymor, "CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets", Work in Progress, draft-ietf-core-coap-tcp-tls-03, July 2016.

[CoRE-RD] Shelby, Z., Koster, M., Bormann, C., and P. Stok, "CoRE Resource Directory", Work in Progress, draft-ietf-core- resource-directory-08, July 2016.

[CRIME] Wikipedia, "CRIME", May 2016, <https://en.wikipedia.org/w/index.php?title=CRIME&oldid=721665716>.

[ENISA-Report2013]
ENISA, "Algorithms, Key Sizes and Parameters Report - <https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-sizes-and-parameters-report>.

[FFDHE-TLS]
Gillmor, D., "Negotiated Finite Field Diffie-Hellman Ephemeral Parameters for TLS", Work in Progress, draft-ietf-tls-negotiated-ff-dhe-10, June 2015.

[HomeGateway]
Eggert, L., Hatoen, S., Kojo, M., Nyrhinen, A., Sarolahti, P., and S. Strowes, "An Experimental Study of Home Gateway Characteristics", In Proceedings of the 10th ACM SIGCOMM conference on Internet measurement,DOI 10.1145/1879141.1879174, 2010,
<http://conferences.sigcomm.org/imc/2010/papers/p260.pdf>.

[IANA-TLS] IANA, "Transport Layer Security (TLS) Parameters",<https://www.iana.org/assignments/tls-parameters>

[ImprintingSurvey]
Chilton, E., "A Brief Survey of Imprinting Options for Constrained Devices", March 2012,

<http://www.lix.polytechnique.fr/hipercom/
SmartObjectSecurity/papers/EricRescorla.pdf>.

[Keylength]
Giry, D., "Cryptographic Key Length Recommendations", September 2015, <http://www.keylength.com>.

[LWM2M] Open Mobile Alliance, "Lightweight Machine-to-Machine Requirements", Candidate Version 1.0, OMA-RD-LightweightM2M-V1_0-20131210-C, December 2013,
<http://openmobilealliance.org/about-oma/work-program/m2m-enablers>.

[PSK-AES-CCM-TLS]
Schmertmann, L. and C. Bormann, "ECDHE-PSK AES-CCM Cipher Suites with Forward Secrecy for Transport Layer Security (TLS)", Work in Progress, draft-schmertmann-dice-ccm-psk-pfs-01, August 2014.

[RFC1981] McCann, J., Deering, S., and J. Mogul, "Path MTU Discovery for IP version 6", RFC 1981, DOI 10.17487/RFC1981, August 1996, <http://www.rfc-editor.org/info/rfc1981>.

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed- Hashing for Message Authentication", RFC 2104,
DOI 10.17487/RFC2104, February 1997,
<http://www.rfc-editor.org/info/rfc2104>.

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, DOI 10.17487/RFC2865, June 2000,
<http://www.rfc-editor.org/info/rfc2865>.

[RFC3610] Whiting, D., Housley, R., and N. Ferguson,

"Counter with CBC-MAC (CCM)", RFC 3610, DOI 10.17487/RFC3610, September 2003, <http://www.rfc-editor.org/info/rfc3610>.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, Ed., "Extensible Authentication Protocol (EAP)", RFC 3748, DOI 10.17487/RFC3748, June 2004, <http://www.rfc-editor.org/info/rfc3748>.

[RFC4086] Eastlake 3rd, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, DOI 10.17487/RFC4086, June 2005, <http://www.rfc-editor.org/info/rfc4086>.

[RFC4492] Blake-Wilson, S., Bolyard, N., Gupta, V., Hawk, C., and B. Moeller, "Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)", RFC 4492,DOI 10.17487/RFC4492, May 2006, <http://www.rfc-editor.org/info/rfc4492>.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, DOI 10.17487/RFC4821, March 2007, <http://www.rfc-editor.org/info/rfc4821>.

[RFC4919] Kushalnagar, N., Montenegro, G., and C. Schumacher, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals",RFC 4919, DOI 10.17487/RFC4919, August 2007, <http://www.rfc-editor.org/info/rfc4919>.

[RFC5077] Salowey, J., Zhou, H., Eronen, P., and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", RFC 5077, DOI 10.17487/RFC5077,January 2008, <http://www.rfc-editor.org/info/rfc5077>.

[RFC5116] McGrew, D., "An Interface and Algorithms for Authenticated Encryption", RFC 5116, DOI 10.17487/RFC5116, January 2008, <http://www.rfc-editor.org/info/rfc5116>.

[RFC5216] Simon, D., Aboba, B., and R. Hurst, "The EAP-TLS Authentication Protocol", RFC 5216, DOI 10.17487/RFC5216, March 2008, <http://www.rfc-editor.org/info/rfc5216>.

[RFC5247] Aboba, B., Simon, D., and P. Eronen, "Extensible Authentication Protocol (EAP) Key Management Framework", RFC 5247, DOI 10.17487/RFC5247, August 2008, <http://www.rfc-editor.org/info/rfc5247>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <http://www.rfc-editor.org/info/rfc5280>.

[RFC5288] Salowey, J., Choudhury, A., and D. McGrew, "AES Galois Counter Mode (GCM) Cipher Suites for TLS", RFC 5288, DOI 10.17487/RFC5288, August 2008, <http://www.rfc-editor.org/info/rfc5288>.

[RFC5480] Turner, S., Brown, D., Yiu, K., Housley, R., and T. Polk, "Elliptic Curve Cryptography Subject Public Key Information", RFC 5480, DOI 10.17487/RFC5480, March 2009, <http://www.rfc-editor.org/info/rfc5480>.

[RFC5758] Dang, Q., Santesson, S., Moriarty, K., Brown, D., and T. Polk, "Internet X.509 Public Key Infrastructure: Additional Algorithms and Identifiers for DSA and ECDSA", RFC 5758, DOI 10.17487/RFC5758, January 2010,

<http://www.rfc-editor.org/info/rfc5758>.

[RFC5934] Housley, R., Ashmore, S., and C. Wallace, "Trust Anchor Management Protocol (TAMP)", RFC 5934,DOI 10.17487/RFC5934, August 2010, <http://www.rfc-editor.org/info/rfc5934>.

[RFC6024] Reddy, R. and C. Wallace, "Trust Anchor Management Requirements", RFC 6024, DOI 10.17487/RFC6024, October 2010, <http://www.rfc-editor.org/info/rfc6024>.

[RFC6090] McGrew, D., Igoe, K., and M. Salter, "Fundamental Elliptic Curve Cryptography Algorithms", RFC 6090,DOI 10.17487/RFC6090, February 2011, <http://www.rfc-editor.org/info/rfc6090>.

[RFC6234] Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234,DOI 10.17487/RFC6234, May 2011, <http://www.rfc-editor.org/info/rfc6234>.

[RFC6655] McGrew, D. and D. Bailey, "AES-CCM Cipher Suites for Transport Layer Security (TLS)", RFC 6655,DOI 10.17487/RFC6655, July 2012, <http://www.rfc-editor.org/info/rfc6655>.

[RFC6690] Shelby, Z., "Constrained RESTful Environments (CoRE) Link Format", RFC 6690, DOI 10.17487/RFC6690, August 2012, <http://www.rfc-editor.org/info/rfc6690>.

[RFC6733] Fajardo, V., Ed., Arkko, J., Loughney, J., and G. Zorn, Ed., "Diameter Base Protocol", RFC 6733,DOI 10.17487/RFC6733, October 2012, <http://www.rfc-editor.org/info/rfc6733>.

[RFC6943] Thaler, D., Ed., "Issues in Identifier Comparison for Security Purposes", RFC 6943, DOI

10.17487/RFC6943, May 2013,
<http://www.rfc-editor.org/info/rfc6943>.

[RFC6961] Pettersen, Y., "The Transport Layer Security (TLS) Multiple Certificate Status Request Extension", RFC 6961, DOI 10.17487/RFC6961, June 2013,
<http://www.rfc-editor.org/info/rfc6961>.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228,DOI 10.17487/RFC7228, May 2014,
<http://www.rfc-editor.org/info/rfc7228>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252,DOI 10.17487/RFC7252, June 2014,
<http://www.rfc-editor.org/info/rfc7252>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014,
<http://www.rfc-editor.org/info/rfc7258>.

[RFC7366] Gutmann, P., "Encrypt-then-MAC for Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", RFC 7366, DOI 10.17487/RFC7366, September 2014,
<http://www.rfc-editor.org/info/rfc7366>.

[RFC7390] Rahman, A., Ed. and E. Dijk, Ed., "Group Communication for the Constrained Application Protocol (CoAP)", RFC 7390, DOI 10.17487/RFC7390, October 2014,
<http://www.rfc-editor.org/info/rfc7390>.

[RFC7397] Gilger, J. and H. Tschofenig, "Report from the Smart Object Security Workshop", RFC 7397, DOI 10.17487/RFC7397, December 2014,
<http://www.rfc-editor.org/info/rfc7397>.

[RFC7400] Bormann, C., "6LoWPAN-GHC: Generic Header Compression for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 7400, DOI 10.17487/RFC7400, November 2014, <http://www.rfc-editor.org/info/rfc7400>.

[RFC7452] Tschofenig, H., Arkko, J., Thaler, D., and D. McPherson, "Architectural Considerations in Smart Object Networking", RFC 7452, DOI 10.17487/RFC7452, March 2015, <http://www.rfc-editor.org/info/rfc7452>.

[RFC7465] Popov, A., "Prohibiting RC4 Cipher Suites", RFC 7465, DOI 10.17487/RFC7465, February 2015, <http://www.rfc-editor.org/info/rfc7465>.

[RFC7507] Moeller, B. and A. Langley, "TLS Fallback Signaling Cipher Suite Value (SCSV) for Preventing Protocol Downgrade Attacks", RFC 7507, DOI 10.17487/RFC7507, April 2015, <http://www.rfc-editor.org/info/rfc7507>.

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <http://www.rfc-editor.org/info/rfc7525>.

[RFC7539] Nir, Y. and A. Langley, "ChaCha20 and Poly1305 for IETF Protocols", RFC 7539, DOI 10.17487/RFC7539, May 2015, <http://www.rfc-editor.org/info/rfc7539>.

[RFC7568] Barnes, R., Thomson, M., Pironti, A., and A. Langley, "Deprecating Secure Sockets Layer Version 3.0", RFC 7568, DOI 10.17487/RFC7568, June 2015, <http://www.rfc-editor.org/info/rfc7568>.

[RFC7748] Langley, A., Hamburg, M., and S. Turner, "Elliptic Curves for Security", RFC 7748, DOI 10.17487/RFC7748, January 2016, <http://www.rfc-editor.org/info/rfc7748>.

[SP800-107-rev1]
National Institute of Standards and Technology, "Recommendation for Applications Using Approved Hash Algorithms", NIST Special Publication 800-107, Revision 1, DOI 10.6028/NIST.SP.800-107r1, August 2012, <http://csrc.nist.gov/publications/nistpubs/800-107-rev1/sp800-107-rev1.pdf>.

[SP800-22b]
National Institute of Standards and Technology, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications", NIST Special Publication 800-22, Revision 1a, April 2010, <http://csrc.nist.gov/publications/nistpubs/800-22-rev1a/SP800-22rev1a.pdf>.

[SP800-90A]
National Institute of Standards and Technology, "Recommendation for Random Number Generation Using Deterministic Random Bit Generators", NIST Special Publication 800-90A Revision 1, DOI 10.6028/NIST.SP.800-90Ar1, June 2015, <http://csrc.nist.gov/publications/drafts/800-90/sp800-90a_r1_draft_november2014_ver.pdf>.

[TLS-FALSESTART]
Langley, A., Modadugu, N., and B. Moeller, "Transport Layer Security (TLS) False Start", Work in Progress, draft-ietf-tls-falsestart-02, May 2016.

[Triple-HS]
Bhargavan, K., Delignat-Lavaud, C., Pironti, A., and P. Yves

Strub, "Triple Handshakes and Cookie Cutters: Breaking and Fixing Authentication over TLS", In Proceedings of the IEEE Symposium on Security and Privacy, Pages 98-113, DOI 10.1109/SP.2014.14, 2014.

附錄 A. 通過 SMS 傳送 DTLS

本節是使用 DTLS 而不是 SMS 的規範。計時器建議已在第 11 節中概述，也適用於通過 SMS 傳輸 DTLS。

本節要求讀者熟悉[GSM-SMS]和[WAP-WDP]中描述的術語和概念。

本節的其餘部分假設行動電台能夠生成和使用編碼為 8 位元二進制數據的傳輸協定數據單元（TPDUs）。

A.1. 概述

DTLS 增加了對 TLS [RFC5246]交握的額外往返，以作為針對某些類型的 DoS 攻擊的保護的返回可路由性測試。 因此，完整的 DTLS 交握包括多達 6 個“行程”（即，邏輯消息交換），然後使用第 4.2 節中描述的分段和重組（SaR）方案將其中的每一個映射到一個或多個 DTLS 記錄。 3 [RFC6347]。 所述方案的開銷是每個交握消息 6 個字節，給定逼真的 10+消息交握，在整個交握序列中將達到大約 60 個字節。

請注意，DTLS SaR 方案僅針對交握消息定義。 事實上，DTLS 記錄從不分段，必須適合單個傳輸層資料封包。

SMS 還提供可選的分段和重組方案，稱為級聯短消息（參見[GSM-SMS]的第 9.2.3.24.1 節）。 但是，由於無法繞過 DTLS 中的 SaR 方案，因此在交握期間不應使用級聯短消息機制來避免冗餘開銷。 在開始交握階段（主動或被動）之前，必須使用 SMS 傳輸的路徑 MTU（PMTU）顯式配置 DTLS 實現，以便正確地檢測其 SaR 功能。 如果使用基於無線資料封包協定（WDP）的複用，則 PMTU 應為 133 字節（見附錄 A.3）；否則為 140 字節。

建議使用在最長可能時段內建立的安全上下文（可能直到接收到關閉警報消息或在非常長的不活動超時之後）以避免昂貴的重新建立安全關聯。

A.2. 訊息分段和重組

SMS 消息的內容在 TP-UserData 欄位中攜帶，其大小可以高達 140 字節。 如附錄 A.1 中已經提到的，可以使用級聯 SMS 發送更長（即，高達 34170 字節）的消息。

該方案消耗 TP-UserData 欄位的 6-7 字節（取決於是使用短分段格式還是長分段格式），從而將可用於 SMS 消息的實際內容的空間減少到每 TPDU 133-134 字節。

雖然原則上可以使用高於 140 字節的 PMTU 值，考慮到其更有效地使用傳輸，這可能看起來像是一種吸引人的選擇，但是存在需要考慮的缺點。 首先，每個 TPDU 需要支付 7 個字節的額外開銷才能支付給 SaR 功能（除了 DTLS SaR 機制引入的開銷之外）其次，一些網路僅部分支援串聯 SMS 功能，而其他網路根本不支援它。

由於這些原因，不應使用串聯短消息機制，並且建議保留在交握階段使用的相同 PMTU 設置，即如果啟用基於 WDP 的多工則保留 133 字節; 否則為 140 字節。

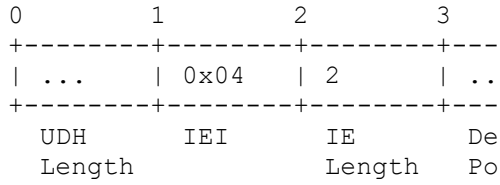
請注意，在 DTLS 交握完成之後，任何與應用層相關的片段和重組邏輯（例如，將 CoAP 消息分段為 DTLS 記錄並在成功執行加密操作後重新組裝它們）需要由使用的應用程序處理已建立的 DTLS 隧道。

A.3. 多工安全關聯

與 IPsec 封裝安全有效負載（ESP）/身份驗證標頭（AH）不同，DTLS 記錄不包含任何關聯識別符。 應用程序必須安排在同一端點上的關聯之間進行多工，當使用 UDP / IP 時，通常使用主機/端口號完成。

如果 DTLS 服務器允許在任何給定時間激活多個客戶端，則可以使用無線應用協定（WAP）用戶資料封包協定[WAP-WDP]來實現不同安全關聯的多工。（WDP 的使用提供了額外的好處，即上層協定可以獨立於底層無線網路運行，從而實現與應用無關的傳輸切換。）

根據是否使用 1 字節或 2 字節端口識別符，編碼 WDP 源和目標端口的總開銷成本是 SMS 內容可用總數的 5 或 7 個字節，如圖 13 和 14 所示。



注：
UDH =用戶數據標頭
IEI =訊息元素識別符

圖 13：應用端口尋址方案（8 位位址）

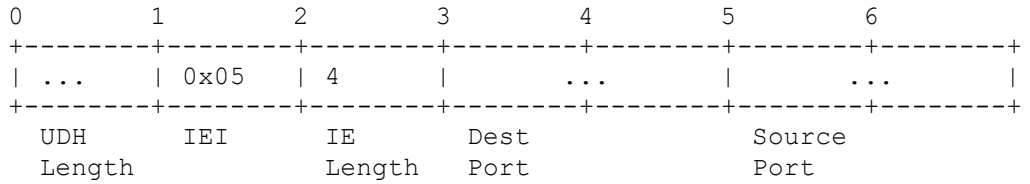


圖 14：應用端口尋址方案（16 位位址）

通信的接收側從 SMS-DELIVER TPDU 的發起者地址（TP-OA）欄位獲得源地址。這樣可以在兩端重建識別安全關聯的唯一 4 元組。（當回復其 DTLS 對等體時，發送方將交換 TP-OA 和目標地址（TP-DA）參數以及 WDP 中的源端口和目標端口。）

A.4. 逾時

如果啟用了 SMS-STATUS-REPORT 消息，則不應將其接收解釋為接收方對特定交握消息採取行動的信號。 因此，DTLS 超時和重傳功能不能將其考慮在內。

交握消息必須攜帶有效期（SMS-SUBMIT TPDU 中的 TP-VP 參數），該有效期不小於重傳超時的當前值。 為了避免在網路中保

留將被接收方丟棄的消息，交握消息應該帶有與重傳超時的當前值相同或略高的有效期。

附錄 B. DTLS 記錄層每封包負擔

圖 15 顯示了使用具有 8 個八位字節完整性檢查值（ICV）的 AES-128-CCM 時，用於保護數據流量的 DTLS 記錄層的負擔。

```
DTLS Record Layer Header.............13 bytes
Nonce (Explicit).....................8 bytes
ICV.................................. 8 bytes
-----------------------------------------------
Overhead.............................29 bytes
-----------------------------------------------
```

圖 15：AES-128-CCM-8 DTLS 記錄層每包負擔

DTLS 記錄層標頭有 13 個八位字節，包括：

- 1 個八位字節內容類型欄位，
- 2 個八位字節版本欄位，
- 2 個八位字節的時間欄位，
- 6 個八位字節序列號，以及
- 2 個八位字節的長度欄位。

AEAD 算法的“隨機數”輸入正好是[RFC5288]的輸入，即 12 字節長。它由兩個值組成，即 4 個八位字節的鹽和 8 個八位字節的 nonce_explicit：

> 鹽是“隱含”的部分，不會在資料封包中發送。相反，鹽是作為交握過程的一部分生成的。
>
> nonce_explicit 值長度為 8 個八位字節，由發送方選擇並在每個 TLS 記錄中攜帶。RFC 6655 [RFC6655]允許 nonce_explicit 是序列號或其他內容。本文使用 DTLS 的限制性更強：64 位 none_explicit 值必須是與 48 位 seq_num 連接的 16 位 epoch。AES-CCM 層上的 nonce_explicit 欄位的序列號分量是記錄層

標頭欄位中序列號的精確副本。 這導致每個記錄重複 8 個字節。

為了避免這種 8 字節重複，RFC 7400 [RFC7400]通過低功耗無線個人區域網路（6LoWPAN）為 IPv6 提供了通用標頭壓縮技術的幫助。 請注意，當通過不使用 IPv6 或 6LoWPAN 的傳輸交換 DTLS 時，此標頭壓縮技術不可用，例如本文附錄 A 中描述的 SMS 傳輸。

附錄 C. DTLS 片段儲存

[RFC6347]的 4.2.3 節建議 DTLS 實現不產生重疊片段。 但是，它要求接收器能夠應對它們。 [RFC6347]的第 4.1.1.1 節解釋了對後者必要性的需求：可以交換準確的 PMTU 估計以縮短交握完成時間。

在許多情況下，處理片段重疊的成本已證明對於受約束的實現是無法承受的，特別是因為緩衝器管理的複雜性增加。

為了減少在同一次握手中產生不同片段大小和隨後重疊的可能性，本文建議：

- 客戶端（交握發起者）將可靠的 PMTU 訊息用於預期目的地; 以及
- 服務器鏡像客戶端選擇的片段大小。
- PMTU 訊息來自客戶端[RFC1981] [RFC4821]執行的“足夠新鮮”的發現或其他一些可靠的帶外通道。

致謝

感謝 Derek Atkins，Paul Bakker，Olaf Bergmann，Carsten Bormann，Ben Campbell，Brian Carpenter，Robert Cragie，Spencer Dawkins，Russ Housley，Rene Hummen，Jayaraghavendran K，Sye Loong Keoh，Matthias Kovatsch，Sandeep Kumar，Barry Leiba，Simon Lemay ，Alexey Melnikov，Gabriel Montenegro，Manuel Pegourie-Gonnard，

Akbar Rahman，Eric Rescorla，Michael Richardson，Ludwig Seitz，Zach Shelby，Michael StJohns，Rene Struik，Tina Tsou 和 Sean Turner，他們提供了對文件形成的有益評論和討論。

非常感謝 Klaus Hartke，他撰寫了本文的初稿。

最後，我們要感謝我們的區域主管（Stephen Farrell）和我們的工作組主席（Zach Shelby 和 Dorothy Gellert）的支援。

作者資訊

Hannes Tschofenig (編輯)
ARM 公司
110 Fulbourn Rd
Cambridge CB1 9NJ
United Kingdom
電子郵件：Hannes.tschofenig@gmx.net
URI:http://www.tschofenig.priv.at

Thomas Fossati
諾基亞
3 Ely Road
Milton, Cambridge CB24 6DD
United Kingdom

電子郵件：thomas.fossati@nokia.com