mbed TLS Introduction

ARM

Simon Butcher

Principal Security Engineer

mbed TLS Tech Lead

Silicon Partner Workshop - Wyboston Lakes
March 2017

Agenda

- What is mbed TLS?
- History of mbed TLS
- Why mbed TLS?
- Applications and Use Cases
- Why trust mbed TLS?
- Features Supported by Hardware Features
- Indicative Code Sizes
- Workshop Outcomes

What is mbed TLS?

- Open Source
  - Available under GPL and Apache licenses
  - Contributions actively encouraged from the community

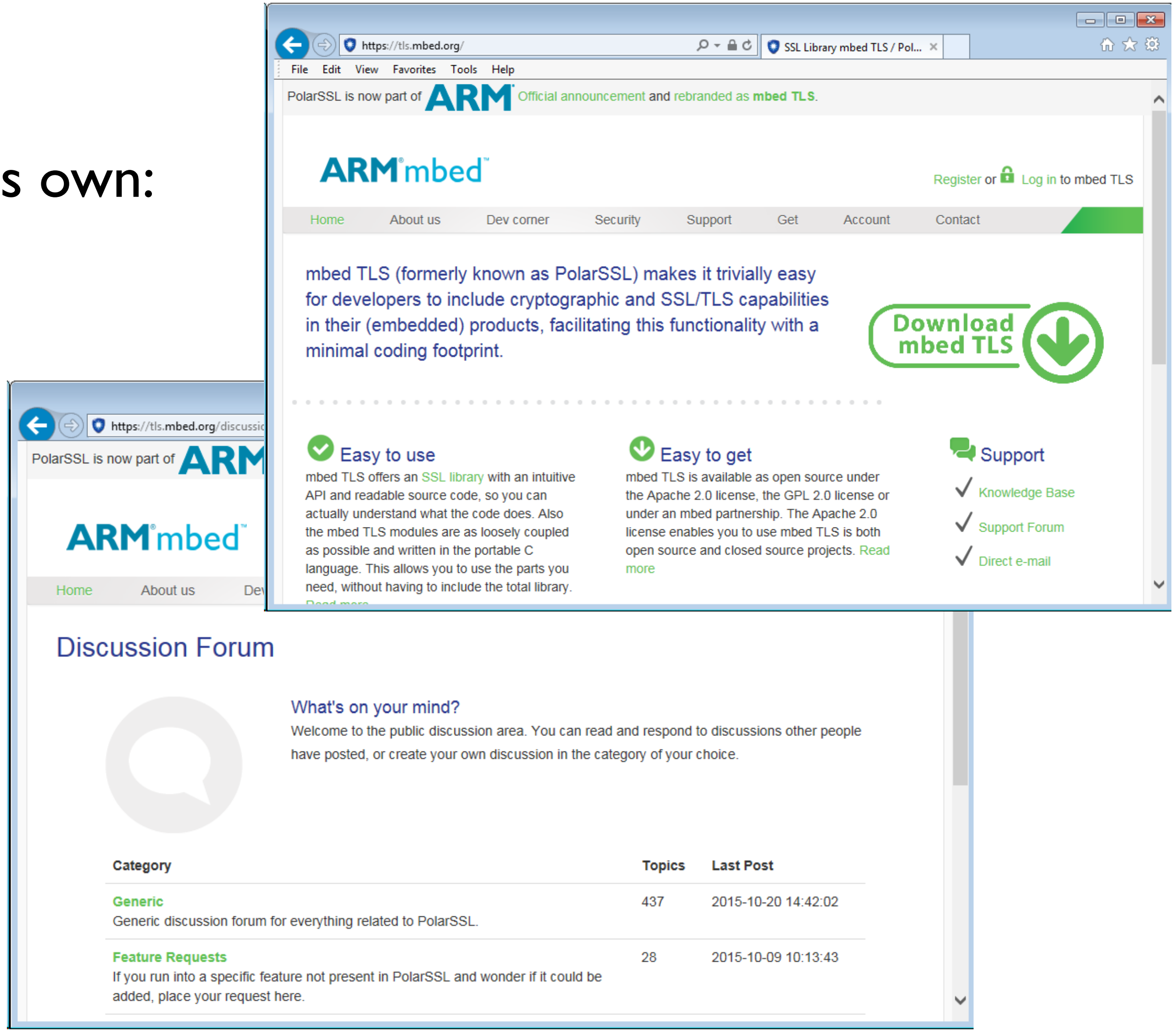
- Self-contained small footprint SSL/TLS library
  - Suitable for embedded and constrained devices, PCs, and servers
  - Highly configurable to reduce footprint to suit required features

- Cryptography library
  - Suitable for use on its own or with the TLS library

- Mature
  - Around for several years, trusted and proven (and subject of many research papers)
  - Strong track record of maintenance and support
  - Ongoing support for previous releases – 1.3 and 2.1(LTS)

mbed TLS as a Product

- mbed TLS is a product in itself, with its own:
  - Customers
  - Releases (independent of mbed OS)
  - Website
  - Support forum
  - Knowledgebase
  - Security Advisories
  - Bug Bounty


https://tls.mbed.org/
SSL Library mbed TLS / Pol...
File Edit View Favorites Tools Help
ARM mbed
Register or Log in to mbed TLS
Home About us Dev corner Security Support Get Account Contact
Download mbed TLS
https://tls.mbed.org/discussi
ARM mbed
Home About us Dev

PolarSSL is now part of ARM Official announcement and rebranded as mbed TLS.

mbed TLS (formerly known as PolarSSL) makes it trivially easy for developers to include cryptographic and SSL/TLS capabilities in their (embedded) products, facilitating this functionality with a minimal coding footprint.

Easy to use

mbed TLS offers an SSL library with an intuitive API and readable source code, so you can actually understand what the code does. Also the mbed TLS modules are as loosely coupled as possible and written in the portable C language. This allows you to use the parts you need, without having to include the total library.

Easy to get

mbed TLS is available as open source under the Apache 2.0 license, the GPL 2.0 license or under an mbed partnership. The Apache 2.0 license enables you to use mbed TLS is both open source and closed source projects. Read more

Support

- ✓ Knowledge Base
- ✓ Support Forum
- ✓ Direct e-mail

PolarSSL is now part of ARM

Discussion Forum

What's on your mind?

Welcome to the public discussion area. You can read and respond to discussions other people have posted, or create your own discussion in the category of your choice.

Category	Topics	Last Post
Generic Generic discussion forum for everything related to PolarSSL.	437	2015-10-20 14:42:02
Feature Requests If you run into a specific feature not present in PolarSSL and wonder if it could be added, place your request here.	28	2015-10-09 10:13:43

History of mbed TLS

- Created in 2006 as XySSL, by Christophe Devine, the Security Researcher famous for Aircrack
- Maintenance passed to Paul Bakker in 2008, who founded OffSpark, renaming it to PolarSSL
- Purchased by ARM in November 2014, and renamed to mbed TLS

Christophe Devine
(Image courtesy SoldierX)

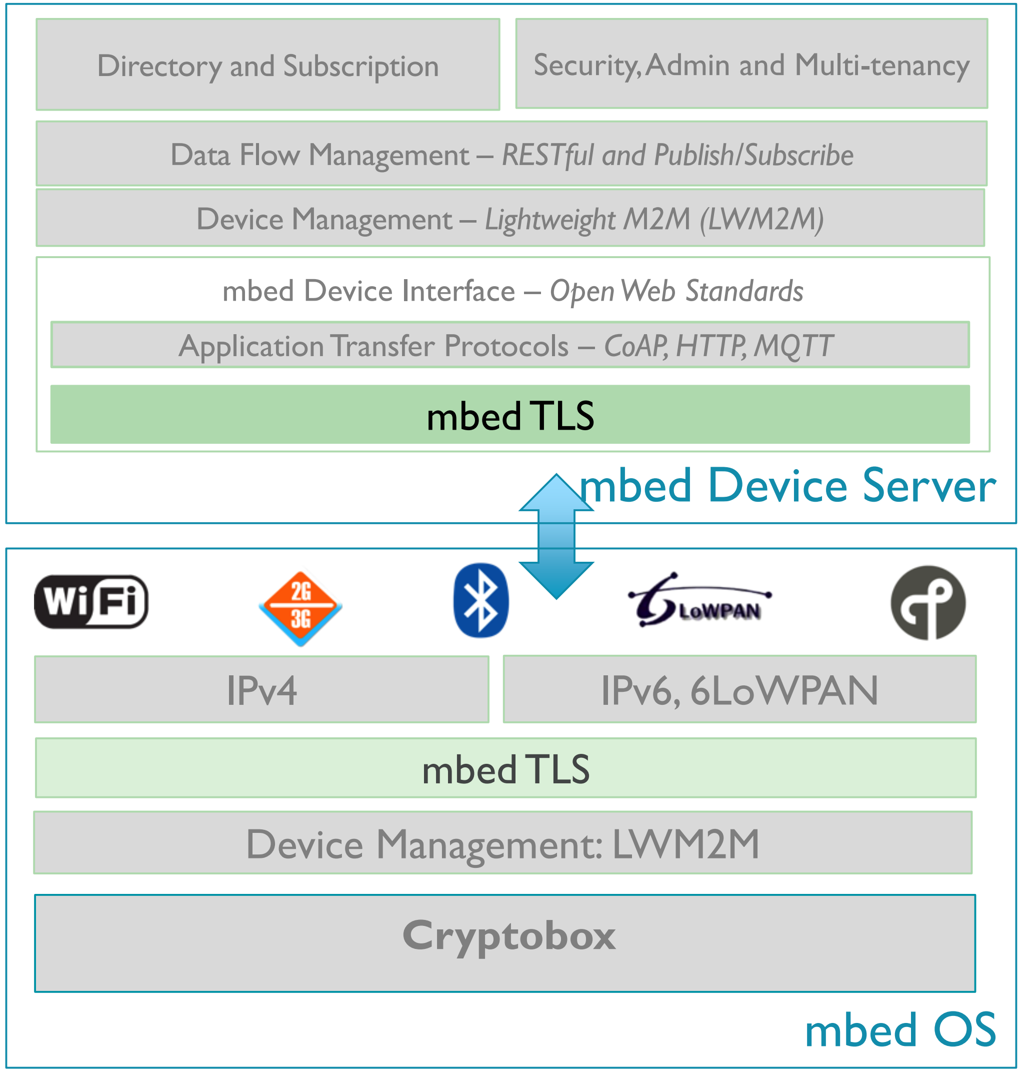
offspark
SPARKING IMAGINATION

Why mbed TLS?

- Leveraging established standards, best practice

- Fully-fledged SSL / TLS / DTLS Library
- Developer friendly: Clean API and documentation
- High assurance: Extensive testing
- Practical: Flexible, small footprint

Transport Security	Symmetric Encryption	Public Key Algorithms	Hash Algorithms	Random Number Generation	X.509 Certificate Handling
TLS/DTLS	AES, 3DES, DES, Blowfish, ARC4, Camellia, XTEA	ECDHE, ECDSA, FFDHE, RSA, FFDH, ECDH	MD2, MD4, MD5, SHA-1, SHA-2, RIPEMD-160	Entropy pool, CTR_DRBG, HMAC_DRBG	✓



Directory and Subscription

Security, Admin and Multi-tenancy

Data Flow Management – *RESTful and Publish/Subscribe*

Device Management – *Lightweight M2M (LWM2M)*

mbed Device Interface – *Open Web Standards*

Application Transfer Protocols – *CoAP, HTTP, MQTT*

mbed TLS

mbed Device Server

IPv4

IPv6, 6LoWPAN

mbed TLS

Device Management: LWM2M

Cryptobox

mbed OS

ARM

Many Possible Use Cases

- Secure communications
  - TLS/DTLS and as a cryptographic library for Thread
- Authentication – cryptographic signing and verification
  - Image verification for secure boot or authenticating patches for update
- Provisioning
  - Remote configuration and secure deployment
- Persistent and volatile data protection
  - Protecting on-device data

mbed TLS Core Features

- Client and Server support of SSL v3, TLS v1.0/v1.1/v1.2, DTLS v1.0/v1.2
- Key exchange methods
    - RSA, FFDHE-RSA, ECDHE-RSA, ECDH-RSA, ECDHE-ECDSA
    - Pre-shared keys – FFDHE-PSK, ECDHE-PSK, RSA-PSK, PSK (no public key)
- NIST Suite B Compliance
- Crypto Algorithms
    - AES, Blowfish, 3DES, DES, ARC4, Camellia, XTEA
    - Modes – ECB, CBC, CFB, CTR, GCM, CCM
    - Hash – MD2, MD4, MD5, SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, RIPEMD-160
    - ECC – NIST, Koblitz, Brainpool curves and Curve25519
    - MAC – HMAC, CMAC
- X.509 reading/writing and validation – PEM/DER formats

How we handle security defects

- Security Researchers and academics occasionally report defects/vulnerabilities
- Such defects or vulnerabilities may not be immediately exploitable
- Significant defects can lead to CVE codes and security advisories
- Security updates and advisories are issued when necessary

Dismantling real-world ECC with Horizontal and Vertical Template Attacks

Margaux Dugardin[1,2], Louiza Papachristodoulou[3], Zakaria Najm[1], Lejla Batina[3], Jean-Luc Danger[1], Sylvain Guilley[1], Jean-Christophe Courrège[2], and Carine Therond[2] *

[1] TELECOM ParisTech, COMELEC, 46 rue Barrault, 75014 Paris, France
firstname.lastname@telecom-paristech.fr
[2] Thales Communications & Security, CESTI, 3 avenue de l'Europe, 31000 Toulouse, France
[3] Radboud University Nijmegen, Digital Security Group, P.O. Box 9010, 6500 GL Nijmegen, The Netherlands
lejla@cs.ru.nl, louiza@cryptologio.org

Abstract. Recent side-channel attacks on elliptic curve algorithms have shown that the security of these cryptosystems is a matter of serious concern. The development of techniques in the area of Template Attacks makes it feasible to extract a 256-bit secret key with only 257 traces. This paper enhances the applicability of this attack by exploiting both the horizontal leakage of the carry propagation during the finite field multiplication, and the vertical leakage of the input data. As a further contribution, our method provides detection and auto-correction of possible errors that may occur during the key recovery. These enhancements come at the cost of extra traces, while still providing a practical attack. Finally, we show that the elliptic curve technology developed in PolarSSL running on a ARM STM32F4 platform is completely vulnerable, when

Features Supported by HW Feature

Entropy Support	Network	Available Features
✗	✗	• Some cryptographic primitives • No key generation • Authentication, not secure communications
✓	✗	• All cryptographic primitives • Key generation • Secure communication, but not TLS
✓	✓	• All cryptographic primitives • Key generation • Full TLS/DTLS Support

Typical Code Size Figures for mbed TLS 2.4.2

- Estimated code size (flash/ROM) figures for current development version of mbed TLS 2.4.2

Configuration Profile*	**Code Size (approx)**
mbed TLS Default	248kb
mbed OS 5.4.1 default	170kb
Thread	58kb
NIST Suite B	80kb
Pre-shared Key	28kb

* *arm-none-eabi-gcc 5.3.1used to obtain figures with flag –Os.*

Workshop Agenda for mbed TLS

Targets for the workshop

- Understand of how to integrate entropy sources
- Understand of how to integrate hardware acceleration support into mbed TLS
- Be able to test and benchmark mbed TLS performance with *your* hardware
- Know how to contribute code changes to mbed TLS
- Know what is expected of contributed code to be accepted

Feedback welcome

- Hardware accelerators come in many different shapes
- mbed TLS API for acceleration and extension is not fixed!
- Feedback on how the API can be adjusted to fit *your* hardware is more than welcome!

Questions?