Mbed TLS 3.0

Overview & Scope

Dave Rodgman & Gilles Peskine

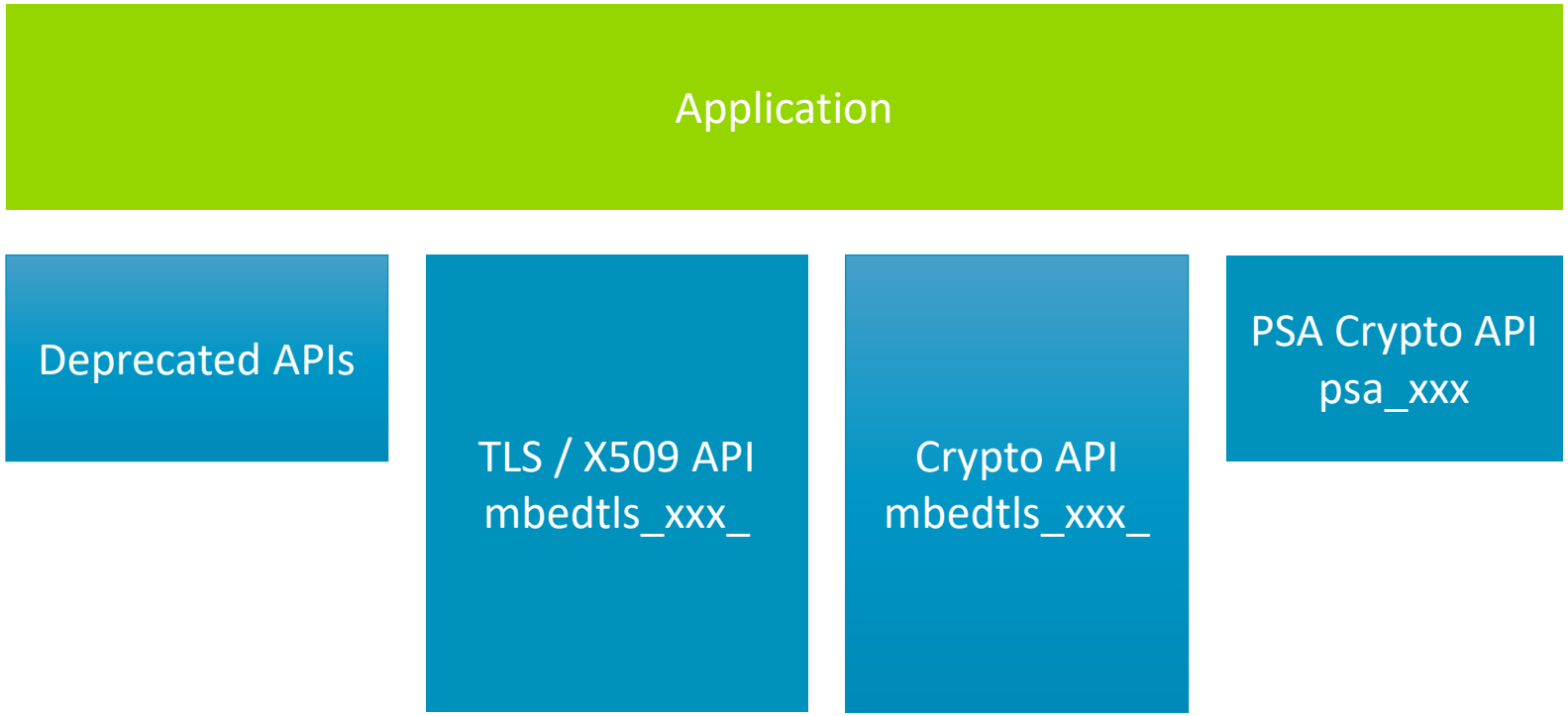
2020-11-03

Agenda

- High-level: current situation, future direction
- Clarify: what is Mbed TLS 3.0
  - And what is it not
  - Some high-level technical details of scope
- What will be the impact for users?
- Support for Mbed TLS 2.x
- Discussion and questions

Mbed TLS 2.x

The situation today

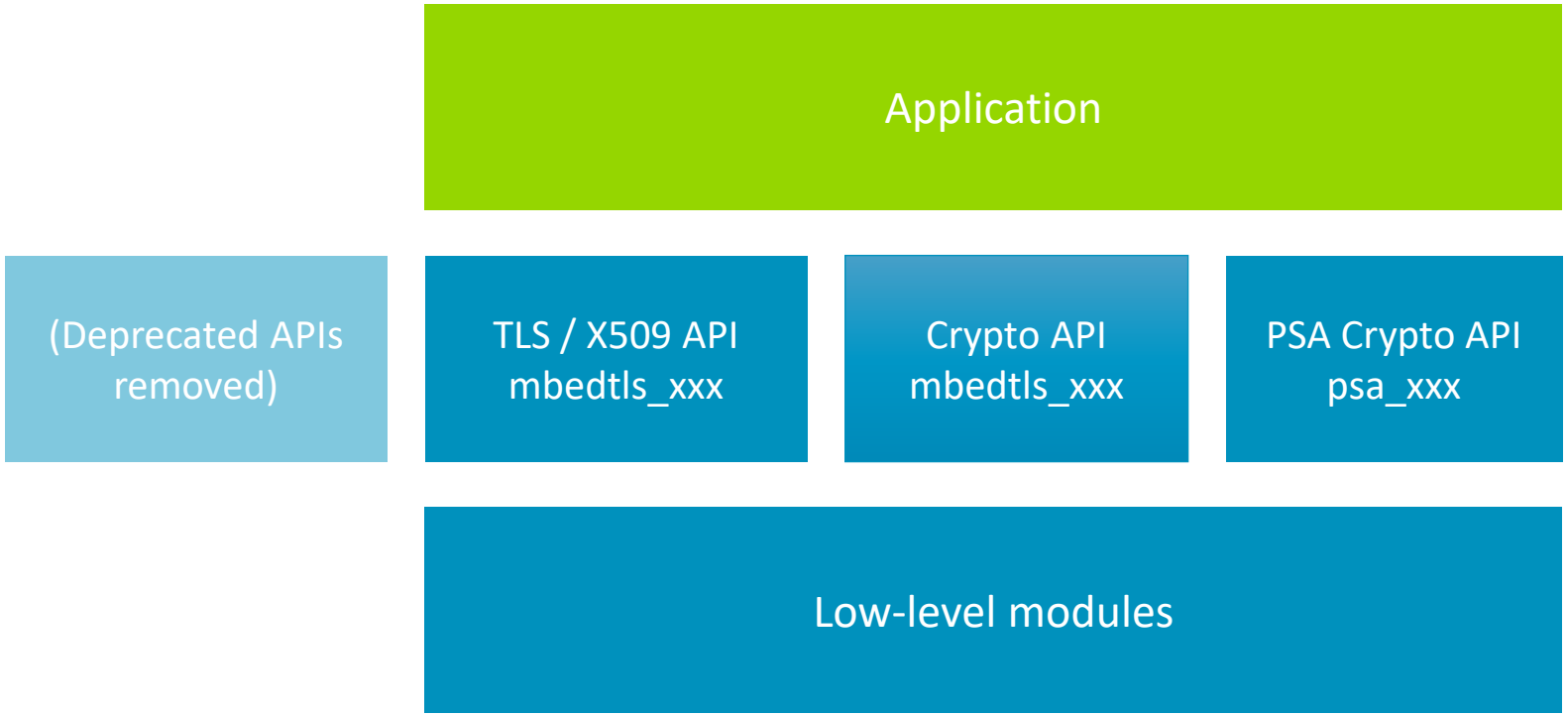


arm

Mbed TLS 3.0

API changes

- Non-backwards compatible
- Remove deprecated APIs
- Hide some low-level Mbed TLS Crypto APIs (accessible via high-level APIs)



arm

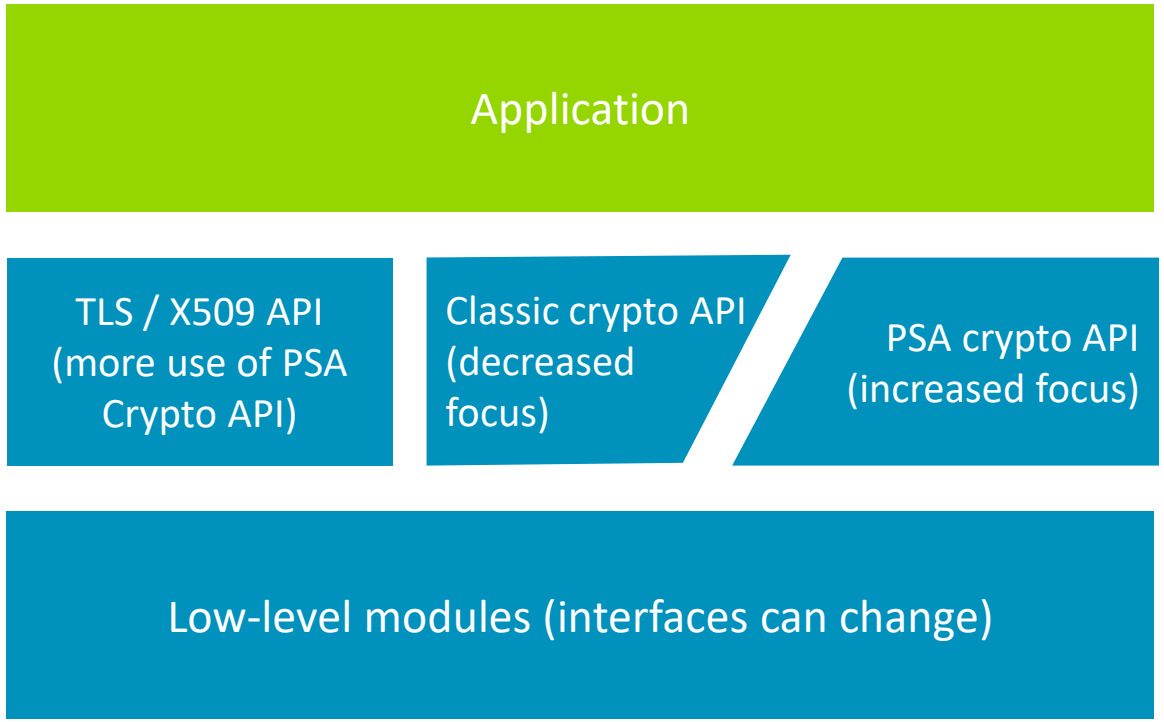
Mbed TLS 3.x

API changes

- Backwards compatible with 3.0

Implementation improvements

- Size reduction, eliminate parts specific to Mbed TLS 2.x



Mbed TLS 3.0 scope

- Feature development not affected
  - TLS 1.3 already starting work in 2.x
  - PSA Crypto extensions in progress in 2.x
- Remove cruft
- Make low-level modules internal
  - Lets us optimize and add functionality that clashes with current low-level design
- Change APIs that are suboptimal in hindsight

Some changes in Mbed TLS 3.0 (not an exhaustive list!)

- Some APIs made private
  - Bignum
  - Various structs become internal
    - Allow us to evolve them over time (and reduce size in some cases)
  - Some low-level crypto functionality
    - Will need to use higher-level APIs instead
- Remove obsolete functionality
  - 74+ deprecated functions
  - Old build options
  - Old TLS options: SSLv3, zlib compression
  - Old TLS cipher suites: RC4, single-DES
- Build changes
  - Split config files: crypto, X509, TLS
- SSL default configuration changes
  - Disable 3DES, non-forward-secure ciphers, CBC ciphers, TLS 1 and TLS 1.1
- Global RNG will be provided
- Many miscellaneous small changes
  - More const pointers
  - init/free API tidy-up
  - Pass buffer sizes consistently
  - …

Impact for users

- Applications on 2.x will be incompatible
  - Some APIs are now non-public
  - Others will change
- Applications using bignum will need an alternative
- 2.x will be maintained alongside 3.x as an LTS branch
  - 2.x will be limited to fixes
  - May consider must-have features for 2.x but unlikely to be accepted

Migration path

- Various options
  - Documentation describing differences and changes required
  - Header file which can help adapt to the 3.0 API
  - Script which can automatically update application code (i.e. similar to 2to3.py)

- Likely that some options will be implemented as part of 3.0 work
  - And others left to community to provide if desired

Timeline

- Will make a 2.x LTS release around the time of the 3.0 release
  - Supported for three years after that
- A 3.x preview branch will be available for a few months
  - The preview branch will not have stable APIs
  - API changes will be discussed on the mailing list (mbed-tls@lists.trustedfirmware.org)
- Users interested in new features should plan to move to 3.x
- Users who need only fixes for a few years could stay on 2.x

Discussion & questions

arm

Thank You
Danke
Merci
谢谢
ありがとう
Gracias
Kiitos
감사합니다
धन्यवाद
شكرًا
ধন্যবাদ
תודה