GitHub - ARMmbed/mbed-crypto: The development of Mbed Crypto has moved to Mbed TLS. No updates will be made to the mbed-crypto repository anymore.

By danh-arm

Archived: 2026-04-05 20:12:44 UTC

The development of Mbed Crypto has moved to Mbed TLS. No updates will be made to the mbed-crypto repository anymore.

Mbed TLS and Mbed Crypto have the same APIs, and the same build system, so most users only need to change the URL to clone or download the library from.

To save build time and possibly avoid system dependencies, you may want to exclude the X.509 and TLS parts of the library by running

or

```
scripts/config.py crypto_full
```

before building Mbed TLS.

See #374 for more details on this migration.

PSA cryptography API

The cryptography library in Mbed TLS is a reference implementation of the cryptography interface of the Arm Platform Security Architecture (PSA). This is a preview release of Mbed Crypto, provided for evaluation purposes only.

Arm's Platform Security Architecture (PSA) is a holistic set of threat models, security analyses, hardware and firmware architecture specifications, and an open source firmware reference implementation. PSA provides a recipe, based on industry best practice, that allows security to be consistently designed in, at both a hardware and firmware level.

The PSA cryptography API provides access to a set of cryptographic primitives. It has a dual purpose. First, it can be used in a PSA-compliant platform to build services, such as secure boot, secure storage and secure communication. Second, it can also be used independently of other PSA components on any platform.

The design goals of the PSA cryptography API include:

- The API distinguishes caller memory from internal memory, which allows the library to be implemented in an isolated space for additional security. Library calls can be implemented as direct function calls if isolation is not desired, and as remote procedure calls if isolation is desired.
- The structure of internal data is hidden to the application, which allows substituting alternative implementations at build time or run time, for example, in order to take advantage of hardware accelerators.
- All access to the keys happens through handles, which allows support for external cryptoprocessors that is transparent to applications.
- The interface to algorithms is generic, favoring algorithm agility.
- The interface is designed to be easy to use and hard to accidentally misuse.

Arm welcomes feedback on the design of the API. If you think something could be improved, please discuss it on the `psa-crypto` mailing list. Alternatively, if you prefer to provide your feedback privately, please email us at `mbed-crypto@arm.com`. All feedback received through this email is treated confidentially.
