COSSACK LABS

Themis

A SINGLE LIBRARY FOR CONSISTENT DATA SECURITY ACROSS ALL PLATFORMS

Implementing cryptography in cross-platform applications is very hard. Choosing cipher suites, defining key lengths, and designing key exchange schemes require plenty of particular competences, and lead to mistakes when done by application developers.

Themis is an open-source **high-level cryptographic services library** for mobile and server platforms. Themis provides ready-made building blocks ("**cryptosystems**") for secure data storage, message exchange, socket connections, and authentication.

Themis brings unified cryptographic security across multiple platforms and is suitable for building sophisticated data security systems. We are using Themis as a core library for our other security products.

Themis contains **4 core cryptographic systems** that meet most of the needs modern applications have towards data security.

Secure Cell for secure storage: container for protecting stored data using symmetric secret. Use Secure Cell to encrypt data at rest: from API tokens to database records.

Secure Comparator for zero-leakage authentication: zero-knowledge proof-based protocol for authentication and handling requests that contain sensitive data, without exposing secrets to the network.

Secure Message for exchanging messages: public key container for sending encrypted and signed data between two parties, to prevent MITM attacks and avoid single secret leakage.

Secure Session for network exchange: session-oriented encrypted data exchange with forward secrecy for protecting sequential data exchanges (API, sessions, chats, sockets).

KEY BENEFITS

Real-world cryptography

Solves 90% use cases for protecting data in mobile and backend apps.

100% compatible API

Fits perfectly for multi-platform apps (mobile, web, server).

Application-level encryption

Strong, audited, tested cryptography for your applications.

Easy to use, hard to misuse

Hides cryptographic details, gives simple building blocks.

Get to market quickly

Themis prevents devs from making security mistakes, saving their time.

Recommended by OWASP

Themis is used by numerous apps, and is recommended by security guidelines.

TYPICAL USE CASES

Encrypt stored secrets in your apps and backend: API keys, session tokens, files.

Encrypt data fields before storing in the DB ("field-level encryption").

Build end-to-end encryption schemes: encrypt data locally on one app, use it encrypted everywhere, decrypt only for authenticated user.

Exchange secrets securely: share sensitive data between parties, build simple chat app between patients and doctors.

LANGUAGES AND PLATFORMS

Core library: C/C++.

Mobile: iOS (Swift, Objective-C), Android (Java, Kotlin), React Native.

Server-side: WASM, Rust, Ruby, Python, Node.js, Go, Java, PHP.

OS: Linux (x86/ARM), macOS, iOS, Android, Windows (experimental).

Ports: Chrome, Redis, PostgreSQL.

Feature documentation and example apps are available for all supported languages and platforms.

License: free, Apache 2 license. **Custom protocols design, integration assistance, and support contracts available depending on your use case.**

SECURE CELL

Secure Cell provides protection of the stored data, such as API tokens, database records or files. Secure Cell uses symmetric encryption with AES256 in GCM or CTR modes.

Secure Cell can be used in 3 different modes:

- **Seal mode** provides the strongest security guarantees and prevents data tampering. AES-256-GCM.
- **Token protect mode** is length-preserving — encrypted objects don't change lengths, yet their authentication data has to be stored elsewhere. AES-256-GCM.
- **Context imprint mode** is a length-preserving mode that contains no authentication tag data. AES-256-CTR.

Secure Cell provides key API and passphrase API. First uses **an embedded key derivation function (KDF),** second uses a **password-based KDF**, so you can use reasonable long keys or passphrases as an argument.

SECURE SESSION

Secure Session is a sequence dependent, stateful messaging session system. It works best for P2P communication with preserved session state, i.e. web sockets or API sessions. Secure Session is decoupled and independent from any networking implementation. It is protocol-agnostic and operates on the 5th layer of the OSI model.

Secure Session is **lightweight**, easy to use and features:

- secure end-to-end communication with perfect forward secrecy & replay protection;
- strong mutual peer authentication;
- low negotiation round-trip;
- strong encryption (ECC + AES);
- straightforward integration process.

Secure Session is stateful and requires session negotiation before data exchange. **Session negotiation** is carried out via **ECDH** with additional security measures against MitM. **Data exchange** is a process of encrypting source data into datagram and sending it to a remote party.

Secure Session can be used via callback API or buffer-aware ("data only") API methodology.

SECURE MESSAGE

Secure Message provides a simple way to protect your messages and bind them to the credentials of communicating peers using strong cryptography. It adds confidentiality, integrity, and authenticity to your message in one shot (as a stateless single function call).

To encrypt the payload, Secure Message uses Secure Cell primitive. Secure Message can work in **signature** and **encryption** modes.

Cryptographic primitives used:

Mode	**Crypto stack**
Elliptic Curve: sign	NIST P-256 + ECDSA
Elliptic Curve: encrypt	NIST P-256 + Secure Cell
RSA: sign	RSA + PSS + PKCS#7
RSA: encrypt	RSA + PKCS#7 + Secure Cell

SECURE COMPARATOR

Secure Comparator allows the parties to prove that they both share some secret — a password, a secret request identifier, or any other verification data:

- **zero data is leaked**: no data which could enable to reconstruct/replay the secret is transmitted;
- **protection against dishonest verifiers**: a verifier can't collect sufficient data to reuse it for further authentication.

Cryptographic primitives used:

Core protocol	Socialist Millionaire's Protocol
Computations	Armoured ed25519
Hash	SHA-256

DOCUMENTATION

docs.cossacklabs.com/themis/