Cramium MPC SDK: Key Features and Roadmap

Sung Hyun Jo, May 5, 2025

Introduction

This document outlines the key features of the Cramium MPC SDK, including available functionalities, ongoing improvements, and upcoming developments. Some features are currently in planning or development.

Currently Available Features

Supporting Multiple MPC Algorithms for Optimal User Experience

Supporting a diverse set of MPC (Multi-Party Computation) algorithms is essential to deliver a flexible and optimized user experience. No single MPC protocol is ideal across all environments, as each involves trade-offs in performance, communication overhead, security assumptions, and network conditions. For example, some algorithms are better suited for low-latency scenarios, while others prioritize minimal computational requirements. Given the wide variability in user scenarios—ranging from mobile wallets with constrained processing power to devices operating under limited network bandwidth—flexibility in algorithm selection is critical.

ECDSA is the dominant digital signature scheme, used by approximately 90% of major blockchains, including Bitcoin and Ethereum. The remaining 10%—typically newer blockchains such as Solana and Cardano—employ alternative signature schemes like EdDSA, Schnorr, or BLS. Unfortunately, ECDSA is a nonlinear signature and MPC protocols for ECDSA are more complex and involve greater trade-offs, typically falling into two primary categories:

- **Homomorphic encryption-based protocols** (e.g., GG18/20, CGGMP21): These favor low communication overhead at the cost of high computational complexity. They are well-suited for environments with sufficient processing capacity, such as modern mobile devices or servers.

- **Oblivious transfer (OT)-based protocols** (e.g., DKLs19, DKLs23): These offer lightweight computation with higher communication demands, making them ideal for resource-constrained environments like web browsers.

The Cramium SDK currently supports the following MPC algorithms:

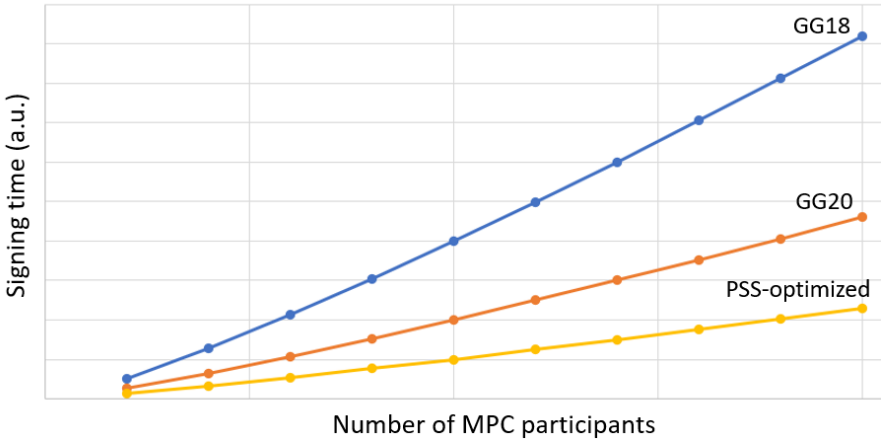
- **ECDSA**: Modified GG18/20 and DKLs23
- **EdDSA, Schnorr**: FROST

As of May 2025, algorithm selection is static and must be determined at deployment time. However, our roadmap includes support for dynamic MPC algorithm selection at runtime. This will enable the system to automatically adapt to the user's operating environment—for example, choosing modified GG18/20 for mobile + cloud settings, or DKLs23 for browser-based scenarios—based on predefined conditions or performance metrics.

Algorithm Enhancements and Security Hardening

Widely adopted MPC algorithms are generally designed for broad use cases, but over time, vulnerabilities and inefficiencies can emerge. At Cramium, we proactively identify and address such issues by continuously auditing and improving these algorithms. For instance, we have patched several known vulnerabilities in GG18/20 to enhance their robustness.

Beyond security hardening, Cramium also tailors algorithmic optimizations to specific usage scenarios. A typical MPC protocol requires the generation and management of multiple secrets (e.g., nonces, private keys, commitments), which can introduce latency and computational overhead. To address this, Cramium developed a Packed Secret Sharing (PSS) scheme that eliminates the need for separate nonce exchange phases and significantly reduces the complexity of homomorphic aggregation. This innovation improves the performance of traditional algorithms such as GG18/20 and Lindell20 by over 50%.



Mnemonic (BIP32/39) Compatibility in MPC

Most cryptocurrency wallets follow the BIP32/39 standards, where private keys are deterministically derived from a master key using hierarchical deterministic (HD) key derivation. This process includes both *hardened* and *non-hardened* derivations. Hardened derivation, in particular, requires the full private key as input to an HMAC function—posing a fundamental compatibility challenge for MPC-based wallets, where private keys are never reconstructed to preserve security.

While garbled circuit techniques can theoretically support hardened derivation without exposing the private key, these methods are computationally expensive, impractically slow, and generally limited to 2-of-2 setups, making them unsuitable for real-world use. As a result, most MPC wallets are not fully compatible with traditional BIP32/39 wallets. Specifically, they cannot import keys from existing mnemonic-based wallets or export keys to other wallets, as the derivation process in MPC differs significantly. This lack of interoperability is a major barrier to broader user adoption.

To address this challenge, Cramium has developed and now offers a **Mnemonic-Compatible MPC** solution alongside its full MPC implementation. Cramium MPC SDK allows a user to have full MPC and/or mnemonic-compatible MPC. Below is a summary of the three key approaches:

- **Verifiable Secret Sharing (VSS):** Although not technically an MPC method, VSS is often marketed as such by some providers. In this approach, a single entity generates the full master private key (or mnemonic), splits it into shares, and distributes them. During signing, shares are gathered and recombined on one device (e.g., a mobile phone or a cloud service) to produce the signature. VSS offers *security at rest* but not *security in-use*, as the key is reconstructed during signing.
- **Full MPC:** Keys are generated through Distributed Key Generation (DKG), such as Feldman's scheme, ensuring that the private key is never fully constructed—even during signature generation. This provides both *security at rest* and *security in-use*, depending on the configuration. However, due to hardened derivation constraints, full MPC is not natively compatible with BIP32/39 and therefore struggles with interoperability.
- **Mnemonic-Compatible MPC (VSS-once + MPC):** Cramium's hybrid solution begins with a single party (e.g., a mobile device) generating mnemonic words and deriving target keys (e.g., for Bitcoin, Ethereum, Solana). The mnemonic and derived private keys are then split into shares and securely distributed. Once this one-time setup using VSS is complete, all subsequent operations—including signing—are performed

using MPC, without reconstructing the key. This approach offers BIP32/39 compatibility, enhances usability across wallet ecosystems, and balances strong security guarantees with practical adoption needs.

Key Rotation and Resharing

While MPC offers significantly stronger security than traditional BIP32/39 wallets, periodic key (or share) rotation remains a recommended best practice for enhanced security assurance. The Cramium SDK supports key rotation, allowing developers to define rotation policies either statically or dynamically based on wallet logic and application needs.

Key resharing—such as adding or removing MPC participants or adjusting the MPC configuration—enables a truly flexible T-of-N threshold system without requiring wallet migration. When combined with an automated backup and recovery scheme, users can seamlessly recover their wallet even if a mobile device, hardware wallet, or other participating device is lost.

Transaction Policy

The Transaction Policy, enforced by the Cramium MPC server, is designed to enhance user security by restricting potentially unauthorized operations through configurable controls and security delays. This feature is optional and can be enabled at the user's discretion.

- **Whitelisting with Security Delay:** When enabled, users are restricted to sending assets only to pre-approved (whitelisted) addresses. Adding a new address to the whitelist requires an activation delay—typically between 24 to 72 hours. This security delay provides a critical window during which users can react and intervene if a malicious actor attempts to add an unauthorized address.
- **Spending Limits:** Users can define spending limits either on a per-transaction basis or as a daily cap, denominated in USD. Once enabled, transactions exceeding the specified limit are blocked. Any changes to the spending limits are subject to a security delay before taking effect, ensuring protection against unauthorized modifications.

Planned Features – Q3 2025

The following features are currently under active development, with completion targeted by the end of Q3 2025. Please note that certain features may be adjusted based on partner requests and evolving priorities.

Support for Additional Blockchains

Cramium is committed to broadening its blockchain support to meet evolving user and market demands. Chain integrations are being rolled out in a phased approach.

- **Currently Supported:**
    - Bitcoin, Ethereum and major Layer 2 networks, BNB Smart Chain, Solana, XRP, Tron, and Ton
- **Upcoming Chain Support (Phased Rollout):**
    - Phase 1: Polygon, Cardano, Avalanche, Dogecoin, Litecoin
    - Phase 2: Aptos, Cosmos, Bitcoin Cash, Stellar, Near Protocol, Internet Computer
    - Phase 3: VeChain, Hedera, Algorand, EOS

EIP-7702 Support

EIP-7702 is an Ethereum Improvement Proposal that introduces a new transaction type, enabling Externally Owned Accounts (EOAs) to temporarily adopt smart contract code. This allows EOAs to benefit from advanced functionalities such as transaction batching, gas sponsorship, and enhanced security features. However, EIP-7702 introduces two potential single points of failure:

- **Security of the Delegated Smart Contract:** Since the EOA delegates authority to an external smart contract, the security of that contract becomes critical. It must be thoroughly audited or widely trusted to mitigate risks.
- **EOA Private Key Authority:** A key distinction between EIP-7702 and EIP-4337 is in the control model. While EIP-4337 embeds all authentication and authorization logic within the smart contract, EIP-7702 retains protocol-level authority with the EOA's private key. This key can unilaterally set or revoke the delegation designator,

regardless of the delegated contract's logic, effectively making it a "super-admin." If compromised, there is no mechanism within the delegated contract to prevent or limit misuse.

How Cramium MPC Addresses These Risks:

- **Smart Contract Security – Address Whitelisting:** Cramium MPC will support an optional address whitelisting for smart contract delegation on a per-chain basis. Each MPC participant will independently manage and verify the whitelist prior to signing, preventing scenarios where a malicious party introduces an unauthorized delegate address.
- **EOA Private Key Protection – MPC Integration:** Cramium will support MPC-based signing for both the authorization tuple (as the authorizer) and the EIP-7702 transaction itself (type 0x04) when the transaction sender and authorizer are the same entity. This mitigates the risk of private key compromise by distributing signing authority across multiple secure parties.

Browser Extension Support

Cramium will offer an MPC SDK not only for mobile and cloud environments but also for browser extensions, enabling a more customizable and flexible MPC deployment. This comprehensive SDK support empowers developers to tailor implementations to diverse user environments and threat models—ranging from on-device computation for mobile users, to cloud-assisted coordination for high availability, to lightweight, installation-free browser-based execution. This enhances both user experience and security adaptability across platforms.

Dedicated Hardware Wallet Support

Expanding the range of supported MPC shareholders enables greater user customization and stronger user-controlled asset security. Cramium will offer a dedicated hardware-based MPC wallet, allowing users to configure MPC setups using any combination of mobile devices, the MPC cloud, web browsers, and hardware wallets. For example, a user could define a 2-of-3 MPC configuration involving a mobile phone, a browser extension, and a hardware device.

Cramium has identified two critical limitations in most existing hardware wallets:

- **Use of Off-the-Shelf General-Purpose MCUs:** Nearly all current hardware wallets rely on off-the-shelf microcontrollers (e.g., from STMicroelectronics), which perform

(partial) cryptographic operations at the firmware level. This lack of direct hardware acceleration leads to suboptimal performance and weaker security assurances.

- **Insecure Traditional Nonvolatile Memory (NVM):** These MCUs typically use charge-based NVM technologies like EEPROM or Flash (e.g., NOR Flash) to store sensitive data. Such memories are inherently vulnerable to physical extraction attacks using techniques like advanced electron microscopy, which can partially or fully reveal stored contents.

To address these challenges, Cramium—backed by a highly experienced IC design team—has developed a custom secure core (or secure element) purpose-built for blockchain applications. This secure core features:

- Direct hardware acceleration for cryptographic operations
- Confidential data storage using ReRAM, the most secure form of nonvolatile memory
- Fabrication at advanced semiconductor nodes (e.g., 22nm), placing it several generations ahead of competitors still using legacy processes (e.g., 40nm)

In addition to supporting MPC workflows, Cramium's hardware wallet is FIDO2-certified and offers secure authentication as well as general-purpose data storage. The Cramium MPC SDK will fully support seamless integration with this dedicated hardware wallet, offering users a highly secure and flexible option for managing their digital assets.

Features Under Consideration

While the features outlined in this section are not currently under active development, Cramium is seriously evaluating them for future support. Upon partner request, prioritization and expedited development may be considered.

Fully Self-Custodial MPC

Cramium is exploring a fully self-custodial MPC architecture in which all key shares are held exclusively on user-controlled devices, such as mobile phones and web browsers—without reliance on the service provider's cloud infrastructure. This approach offers several critical advantages:

- **Enhanced Privacy and Security**: By removing the provider's involvement in key custody and usage, the system eliminates centralized points of failure, significantly reduces the risk of compromise, and maximizes privacy.

- **Reduced Liability**: The service provider is not responsible for any portion of the user's private key, minimizing potential legal and operational liability.
- **Alignment with Decentralization**: This model embraces the principles of self-custody and user sovereignty, making it well-suited for high-security applications and privacy-conscious users.

To support a fully self-custodial experience, Cramium is evaluating several enabling technologies, including direct peer-to-peer (P2P) communication, local QR code exchanges, local-only WebAssembly (WASM) execution, private VPN tunneling, and hybrid combinations of these methods.

Linear Signatures

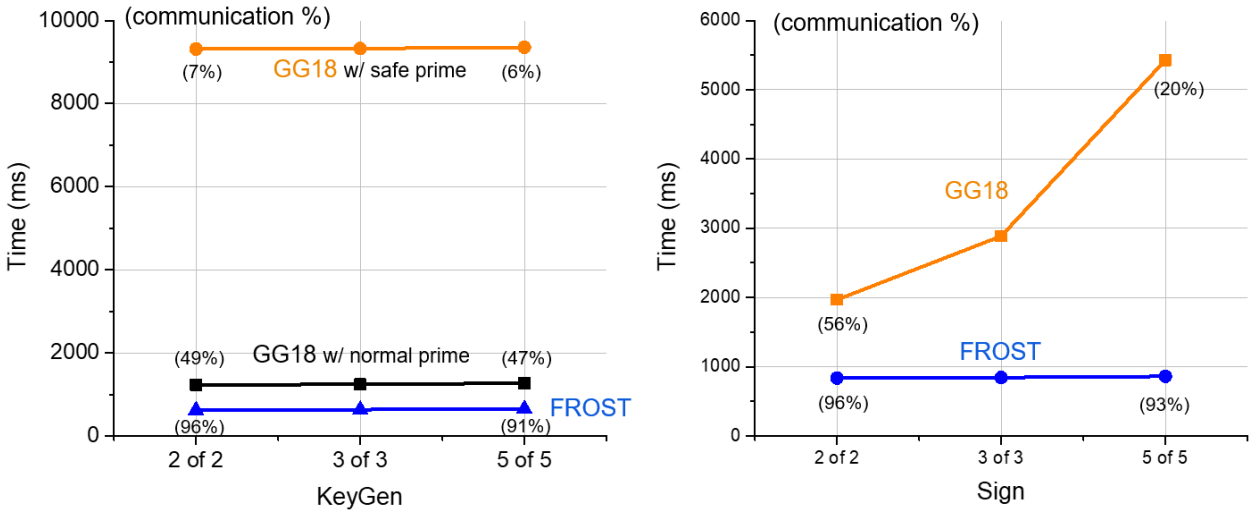
MPC implementations for ECDSA—the signature scheme used by most blockchains—are inherently less efficient due to ECDSA's non-linear structure. In contrast, newer signature schemes such as EdDSA and Schnorr are linear in nature, enabling the development of significantly more efficient and streamlined MPC protocols.

Signature Scheme	Blockchain Example	Category	MPC Algorithm Example
Non-linear signature (e.g., ECDSA)	Bitcoin, Ethereum, BNB Chain	Computing-heavy (homomorphic e.g., Paillier)	GG18/20, CGGMP21
		Communication-heavy	DKLs19, DKLs23
Linear signature (e.g., EdDSA, Schnorr)	Solana, Cardano, Bitcoin (Taproot)	Fast & efficient	FROST, ROAST

Schnorr Signature (linear)	ECDSA Signature (non-linear)
Random $k \leftarrow Z_q$ $R = k \cdot G$ $e = H(m \mid\mid R)$ $s = k - x \cdot e \bmod q$ $sig = (s, e)$ } allows efficient MPC	Random $k \leftarrow Z_q$ $R = k \cdot G$ $r = r_x$, where $R = (r_x, r_y)$ $s = k^{-1} \cdot (H(m) + r \cdot x) \bmod q$ $sig = (r, s)$ } non-trivial for MPC

The figure below illustrates the efficiency advantage of linear signature MPC schemes—such as FROST on Schnorr—compared to non-linear schemes like GG18 on ECDSA. Linear signature MPC enables much more scalable and efficient support for larger *t*-of-*n* configurations. As a mid-term objective, we are considering proposing an EIP to introduce a precompiled contract for Schnorr (or EdDSA) on Ethereum. This would drastically reduce gas

costs for signature verification compared to current smart contract-level implementations, while also enabling the use of linear signatures.



Websites:

- Cramium (Secure IC, Digital Asset Security) https://www.cramiumlabs.com/
- Crossbar (ReRAM) https://www.crossbar-inc.com/