Introducing CDP Wallets (now in Beta): Full Control, Zero Key Management

May 28, 2025

By Yuga Cohler, Josh Nickerson, and Dan Kim

TL;DR *Coinbase Developer Platform is introducing a new kind of developer wallet that gives builders control without touching private keys. Built on Coinbase's secure enclave infrastructure, they let you create, sign, and enforce transaction rules entirely via API - making them ideal for agents, automation, and high-performance onchain systems.*

Developers building onchain have historically faced a tradeoff: maintain full custody of private keys and take on operational complexity, or delegate control to a custodial solution and lose flexibility.

CDP Wallets eliminates that tradeoff, giving developers a new kind of wallet that offers both full control and almost zero operational burden.

CDP Wallets let developers create programmable wallets via API, without ever handling private keys or managing infrastructure. Sensitive wallet functions happen inside a Trusted Execution Environment (TEEs) which ensure that unencrypted private keys are never exposed—not even to Coinbase.

You create, sign, and manage transactions entirely through scoped API calls: no MPC, no key shards, no infrastructure to deploy.

From policy-enforced automation to agentic transactions, CDP Wallets are built for what's next.

A New Type of Developer Wallet

CDP Wallets move beyond the constraints of traditional EOA and MPC-based systems, giving developers full control, stronger security, and none of the operational overhead.

Our previous Wallet API v1, based on MPC, laid important groundwork. But our new CDP Wallets v2 represents a step-change: a fundamentally new architecture purpose-built for automation, scalability, and policy-enforced execution.

Here's how the two generations compare:

FEATURE	CDP WALLETS V1 (EOA/MPC)	CDP WALLETS V2 (TEE)
PRIVATE KEY MANAGEMENT	Developer-managed (EOA) or MPC signer	Encrypted and interactions happen in Trusted Execution Environment (TEE)
SETUP COMPLEXITY	Infra and signer provisioning	API-only, no infra
POLICY ENGINE	None	Built-in, programmable
SMART WALLET SUPPORT	Limited	Gas sponsorship, batching, subaccounts (coming)
KEY RECOVERY / ROTATION	Manual	Secure, built-in

SIGNING PERFORMANCE	Variable	Sub-500ms under typical load
EIP-712 SIGNING	Not supported	Fully supported
SUPPORTED CHAINS	Ethereum, Base, Arbitrum, Polygon	EVM + Solana
DEV SDKS / LIBRARIES	Limited	viem, wagmi, ethers.js, CDP SDKs
USDC REWARDS	No	4.1% Rewards (U.S. developers)

How CDP Wallets Work

CDP Wallets combine enterprise-grade security with developer-first design—built from the ground up to support automation, control, and composability.

Key features

At the core of the system are three foundational features:

Trusted Execution Environments (TEEs)

All sensitive wallet interactions (like decrypting private keys) happen inside an AWS Nitro Enclave, a secure, isolated compute environment where private keys are generated, encrypted, and used for signing. The unencrypted key is never exposed—not even to Coinbase.

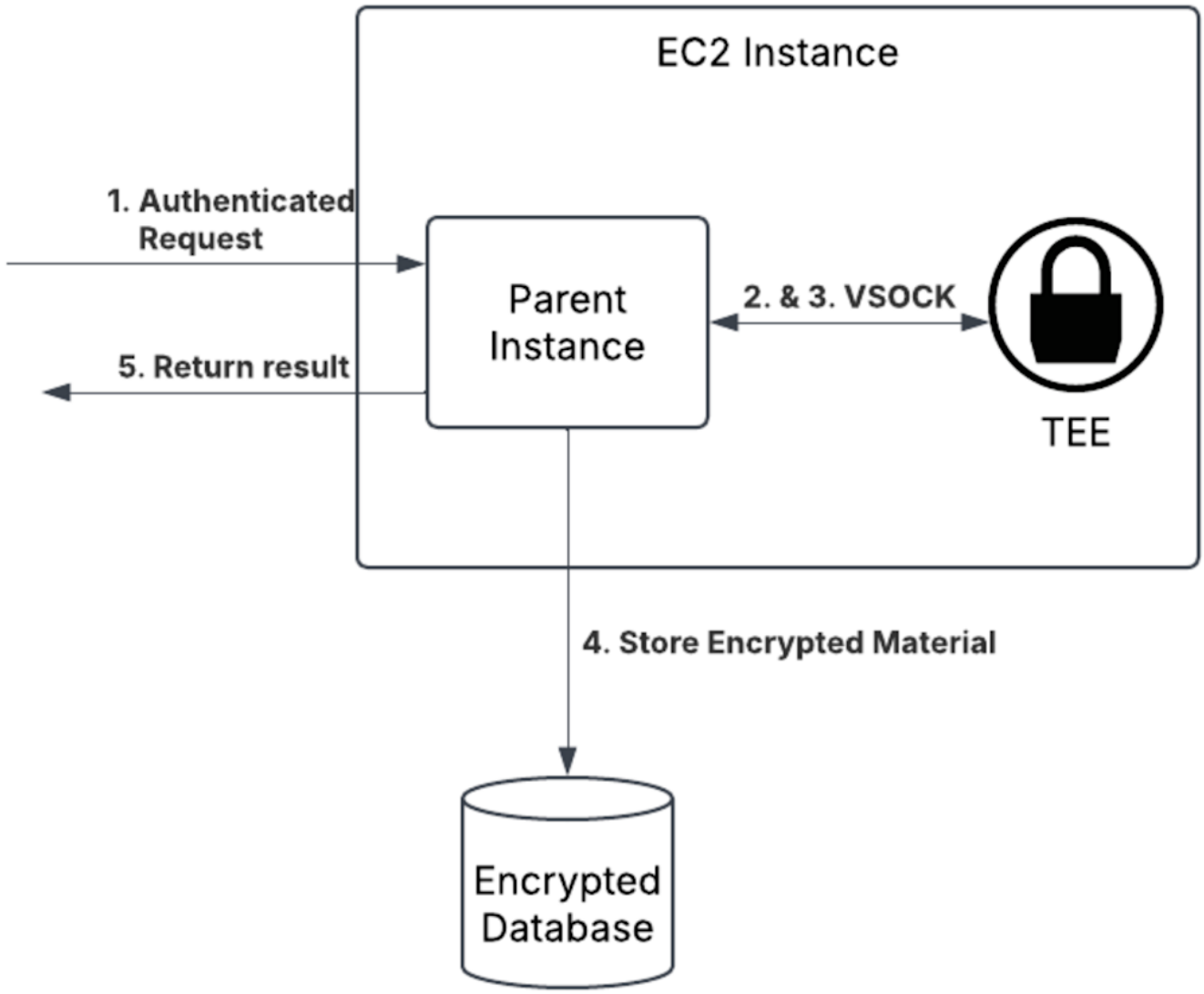
Zero Key Management

Developers never touch key material. Instead, they authenticate with scoped API credentials. There's no need to run signer infrastructure, coordinate key shards, or manage rotation schedules. The result: full control, with none of the operational burden.

Programmable Policy Engine

Developers can define policies that restrict transfers to allowlisted addresses, limit transaction amounts,

block risky destinations using Coinbase KYT, and—coming soon—enforce smart contract-level restrictions. This is essential for things like AI agents. All policies are managed via API or SDK, and enforced at the enclave layer, even in the event of credential compromise.



Tech Specs

Beyond the core architecture, CDP Wallets deliver the performance, integration, and composability developers expect from infrastructure built for automation. Wallets are created in under 500ms, with signing latency typically under 200ms—ideal for real-time automation and high-throughput systems. They integrate directly with viem, wagmi, and ethers.js, and offer SDKs in TypeScript, Python, and Go. CDP Wallets are fully EIP-1193 compliant, and support EVM-compatible chains and Solana.

CDP Wallets also support EIP-712 typed message signing—enabling structured interactions with smart contracts. This powers integrations like x402, where agents authorize payments, services, or execution

onchain—with human-readable verification and policy-enforced safety.

For developers based in the U.S., CDP Wallets also include native USDC rewards: 4.1% rewards on idle

or lockups.

This isn't just wallet infrastructure—it's programmable custody, ready for agents, automation, and the next generation of onchain systems.

What Developers Are Building with CDP Wallets

CDP Wallets can power a wide range of backend wallet operations—but they're especially well-suited for use cases that demand both speed and security. When developers need to move fast, automate execution, and enforce fine-grained controls at the wallet level, CDP Wallets are a natural fit.

Here are five areas where CDP Wallets are already emerging as a go-to choice:

- **AI Agents:** Autonomous agents that trade, send, or interact with smart contracts—safely and under programmable rules.
- **x402-Powered APIs:** CDP Wallets can act as trusted senders in pay-per-use APIs, enabling agents and apps to transact without managing keys.
- **DeFi Bots and Vaults:** Automated systems can rebalance, stake, or harvest yield across protocols using policy-governed wallets.
- **Payments Infrastructure:** Fintech and stablecoin platforms use CDP Wallets for embedded send/receive flows— with KYT and access controls.
- **Enterprise Treasury:** DAOs and teams managing crypto ops use CDP Wallets to segment risk, enforce limits, and automate execution.

Want to see these flows in action? Check out our sample apps and walkthroughs.

Why It Matters

CDP Wallets unlock a new category of infrastructure: wallets that can operate automatically, and at scale —without requiring teams to manage keys, shards, or hosted signers.

If you've built AI agents, automation pipelines, or backend systems, you know the pain: hot keys on servers, custom MPC stacks, or fragmented signer flows. CDP Wallets replace that with an API that offers

both full control and almost zero operational burden.

This is more than a developer convenience—it's a shift in how wallets are used.

The next generation of onchain systems need wallets that are programmable, policy-enforced, and invisible to the end user. CDP Wallets make that possible.

Start Building

CDP Wallets are available now in open beta.

Whether you're developing stablecoin-based payment systems, onchain apps, or agentic workflows, you can start creating secure, self-custodied wallets in minutes—with no infrastructure to deploy.

- Start building
- Explore the docs
- Check out sample apps
- Join the community
- Follow us @coinbasedev

CDP Wallets are built for the programmable onchain era—simple, fast, and ready for production.

We're excited to see what you build—and how far you can take it.

Affiliates
Blog
Press
Security
Investors
Vendors
Legal & privacy
Cookie policy
Cookie preferences
Do Not Share My Personal Information
Digital Asset Disclosures

Learn

Explore
Coinbase Bytes newsletter
Crypto basics
Tips & tutorials
Crypto glossary
Market updates
What is Bitcoin?
What is crypto?
What is a blockchain?
How to set up a crypto wallet
How to send crypto
Taxes

Individuals

Buy & sell
Earn free crypto
Wallet
Card
Coinbase One

Businesses

Asset Listings
Commerce

Institutions

Prime
Staking
Exchange
International Exchange
Derivatives Exchange
Verified Pools

Developers

Developer Platform

Base
Staking
Onramp
Wallets
Wallet SDK
Coinbase App
Exchange API
Prime API
Base Node
OnchainKit

Support

Help center
Contact us
Create account
ID verification
Account information
Payment methods
Account access

Supported crypto
Status