

```
import Figure from "@site/src/components/Figure/Figure"; import LogoGrid from "@site/src/components/LogoGrid/LogoGrid";
```

Wallets are the gateway to Web3, serving as essential portals for users to send and receive messages, manage funds, and interact with blockchain applications. As a critical piece of blockchain infrastructure, wallets significantly shape users' Web3 experiences.

The wallet ecosystem is diverse, with providers offering varied products and services through different mechanisms. As wallet providers strive for sustainability and diversification, their operational models are evolving, creating new dynamics between users, applications, and the underlying blockchain infrastructure.

Our report aims to illuminate the current state of wallets on Ethereum, building upon the research conducted by [orderflow.art](#).

However, identifying wallets through on-chain transaction tracking presents several challenges:

1. Incomplete identification due to unknown routers or signature addresses.
2. Difficulty in identifying multiple wallet addresses associated with centralized exchanges.
3. Absence of router addresses for some wallets, making user identification challenging.

Despite these limitations, this report provides a comprehensive overview of the Ethereum wallet landscape, its current trends, and future prospects.

The report begins with a Background section covering two key areas: Wallet Taxonomy and the Order Flow Lifecycle. In Wallet Taxonomy, we categorize Web3 wallets into custodial and non-custodial types, detailing the various forms of non-custodial wallets. The Order Flow Lifecycle section outlines the journey of a transaction, identifying key players from Order Flow Originators to block builders.

Next, we explore Current Trends, focusing on recent developments affecting Order Flow Originators (OFOs). We examine the implications of increasing centralization in the block building market, which has intensified competition for order flow. This section covers three key concepts: Payment for Order Flow (PFOF), Order Flow Auctions (OFAs), and Private Order Flow (POF). Additionally, we introduce Account Abstraction (AA), with a particular focus on ERC-4337, a significant development reshaping the wallet landscape.

Finally, in Future Trends, we explore developments aimed at enhancing user experience and addressing regulatory challenges in the wallet landscape. We examine pre-confirmations (pre-confs), a mechanism designed to improve transaction confirmation speed. We also analyze two Ethereum Improvement Proposals (EIPs) that seek to enhance Account Abstraction capabilities. Additionally, we discuss Trusted Execution Environments (TEEs) and their role in improving security and privacy for Web3 wallets. We consider how TEEs might serve as a potential compliance solution for the crypto industry, particularly as regulatory focus shifts from decentralization to questions of control.

Background

Introduction

Wallets serve as the primary interface for users to interact with blockchain applications. While users often maintain multiple wallets (for example, several MetaMask accounts), the process of migrating private keys to a new wallet provider is typically cumbersome. This lack of user-friendly portability creates a 'stickiness' effect, often keeping users tied to their existing wallet providers.

The intensifying competition for order flow has heightened the importance of user acquisition and retention for wallet providers. This competitive landscape has led to an interesting development: decentralized finance (DeFi) applications, such as Uniswap, 1inch, and Curve Finance, are now creating their own wallets. This strategic move allows these DeFi platforms to exert greater control over their users' order flow, potentially capturing more value and providing a more integrated user experience.

This trend underscores the evolving relationships between users, wallets, and DeFi applications in the blockchain ecosystem. It highlights how the battle for order flow is reshaping the wallet landscape and influencing the strategies of major players in the DeFi space.

Wallet Taxonomy

Web3 wallets are usually categorized as either custodial, controlled by third parties, or non-custodial, controlled by the user. Control is defined as who holds the private keys to the wallet.

- **Custodial wallets:** Primarily offered by crypto exchanges and Telegram bots, custodial wallets hold users' private keys and offer users an improved user experience. However, users do not have full control over their funds and these third parties could access users' funds without their permission.
- **Non-custodial wallets:** Users hold their private keys and fully control their non-custodial wallets. If the private keys are lost, the user will lose access to their wallet and funds. Private keys for hot wallets are stored on an internet-connected device that interacts with an application. Cold wallets store private keys on dedicated air-gapped hardware devices that do not interact with applications. Non-custodial "hot" wallets are connected to the internet and are usually accessed via a browser extension, mobile app, or desktop application. "Cold" wallets have no online access and assets are stored in physical devices.

There are different types of non-custodial wallets with varying technology to improve user experience and security.

MPC Wallets

[Multi-Party Computation](#) (MPC) wallets use cryptography techniques to encrypt, fragment, and distribute private keys to multiple devices. These devices or parties must evaluate a computation without revealing their private keys or data. A multi-party computation protocol used in the context of MPC wallets usually has these properties:

1. **Threshold Security:** Ensuring that a predefined number of parties must cooperate to sign a transaction.
2. **Key Fragmentation:** The ability to split a private key into multiple shares.
3. **Distributed Key Generation:** Generating the key in a distributed manner so that no single party ever knows the full private key.

The benefits of MPC wallets are:

1. **Security:** Since no single person controls the private key, an attacker would be required to attack multiple parties increasing a wallet's security.
2. **Recoverability:** With encrypted key fragments stored in multiple places, authorized parties can recover accounts if a key is lost.
3. **Accessibility:** Assets can be held online since the private key fragments are securely distributed among multiple parties. Transactions can be executed more efficiently than a hot wallet without compromising key security.

Externally Owned Accounts (EOAs)

Externally Owned Accounts (EOAs) are managed by unique private keys that users control to interact with smart contracts on-chain.

EOAs use a private [Elliptic Curve Digital Signature Algorithm](#) (ECDSA) key to sign and verify digital transactions. Users can send and receive transactions, interact with smart contracts, and approve messages through EOAs.

To create an EOA, a wallet UI generates a private key and a seed phrase. Because of the singular private key and seed phrase, a user will lose access to their wallet if they lose both their private key and seed phrase.

Smart Contract Wallets

Smart contract wallets, or smart wallets, utilize Account Abstraction and the programmability of smart contracts to improve user experience. Smart contract wallets are not controlled by a private key but by the contract code. Account Abstraction protocol like ERC-4337 helps smart contract wallets bypass the requirement that an EOA wallet initiates a transaction. Smart contract wallets can be programmed for features such as:

1. Two-factor authentication
2. Account freezing
3. Flexible recovery
4. Transaction batching
5. Transfer and spending limits
6. Session keys
7. Gas sponsorship and non-native token gas payments
8. Multi-sig wallet

Compared to EOAs, smart contract wallets have a small gas overhead mainly due to the execution of contract code and the publishing of events. Smart contracts are inherently more complex and powerful relative to EOAs, so only audited and battle-tested smart contract wallets should be trusted.

Order Flow Lifecycle

[Orderflow.art](#) illuminated the order flow landscape and identified the known on-chain actors in a transaction's lifecycle.

A transaction's life cycle begins on the left-hand side of the order flow Sankey with on-chain frontends and ends on the right-hand side with block builders.

The key on-chain actors in a transaction's lifecycle are:

Order Flow Originators

Order Flow Originators (OFOs) are the first on-chain applications that interact with a wallet. OFOs include:

1. **Wallets:** Wallets are increasingly adding more functionality to improve user experience such as direct swaps.

Note: Figures 3, 4, and 5 only include known routers and underrepresents the native swap transactions from wallets.

1. **Frontends:** Applications like Uniswap have their own wallet and interface for users to create transactions. DEX frontends are losing dominance in both trading volume and transaction count market share (Figures 3 and 4).
2. **Telegram Bots:** Banana Gun, Maestro, and Unibot have captured a significant portion of retail transactions within the last year (Figure 4). Trade sizes are generally less than \$10,000 (Figure 5).
3. **Aggregators:** Aggregators are applications like DefiLlama, Matcha, 0x API, and 1inch API that connect to several DEXs to unify fragmented liquidity. The transaction counts market share has remained relatively consistent while trading volume has decreased slightly since 2023 (Figures 3 and 4).
4. **Order Flow Auctions:** OFAs include solver batch auctions (e.g., CoWSwap), RFQ systems (e.g., Uniswap X), and execution auctions (e.g., MEV-Blocker). OFAs have been gaining trading volume market share at the expense of DEX Frontends (Figure 3) and are typically used for larger trades (Figure 5).

Liquidity Providers

Large transactions or those involving illiquid trading pairs are often routed to Order Flow Auctions (OFAs) and aggregators to minimize slippage. These providers source liquidity from multiple decentralized exchanges (DEXs), off-chain sources, and proprietary inventories.

1. **Market Makers:** Trading entities that use off-chain liquidity and their own inventory to execute transactions. They provide liquidity for Request for Quote (RFQ) platforms like Hashflow and Uniswap X.
2. **Solvers:** Third-party entities that determine optimal routing and pricing for transaction execution. Solvers are utilized in OFAs such as CoWSwap, and some offer direct user transaction submission through their own front-end interfaces.
3. **CEX-DEX Searchers:** These searchers leverage off-chain liquidity from centralized exchanges (CEXs) to capture on-chain arbitrage opportunities. They can utilize OFAs with private mempools like MEV Share and MEV-Blocker.

Mempools

Ethereum orders are submitted to either public or private mempools:

1. **Public mempools:** Transactions are visible to all and can be picked up for bundling by searchers and OFAs. All block builders can access these transactions for block inclusion.
2. **Private mempools:** Transactions are visible only to select parties, including specific searchers, OFAs, and builders.

Builders

Builders arrange and include transactions in a block. The order's lifecycle is complete if the transaction is included in the winning builder's block. If not included in the winning block, the transaction remains in the mempool until it is either included in a future block or discarded.

References

1. <https://orderflow.art/frontends>
2. Quicknode, [A Complete Overview of Web3 Wallets](#)

3. Ethereum Foundation, [Ethereum Accounts](#)
4. Fireblocks, [What is MPC \(Multi-Party Computation\)?](#)
5. Alchemy, [8 Amazing Benefits of Smart Contract Wallets vs EOA Wallets](#)
6. Unchained, [What are Externally Owned Accounts \(EOAs\) in Ethereum?](#)
7. Ambire Wallet, [Account Abstraction and the Benefits of Smart Contract Wallets](#)

Current Trends

The Ethereum landscape is currently characterized by several significant trends that are reshaping the industry. Two major trends in Ethereum that affect Order Flow Originators are 1) the centralization of the block-building market, and 2) the implementation of Account Abstraction with ERC-4337.

Block Builder Centralization

[Ethereum's builder market](#) has become increasingly centralized with two builders capturing more than 90% of the block market.

This concentration has given rise to new dynamics in order flow:

- Payment for Order flow (PFOF)
- Order Flow Auctions (OFA)
- Private Order Flow

These mechanisms are transforming how transactions are processed and prioritized, offering benefits like MEV protection and improved price discovery, but also raising concerns about market fairness and decentralization

Payment for Order Flow

Payment for Order Flow ("PFOF") is a traditional finance concept that started with market makers paying brokerages for their OTC order flow. Market makers consider retail order flow uninformed and non-toxic and are highly profitable to trade against. As automated trading systems ("ATS") expanded, market makers used PFOF to attract retail order flow to their ATS.

Retail traders benefit from PFOF in three ways:

1. Part of the PFOF is used to cover the retail trader's execution cost.
2. Market makers will quote tighter enabling retail traders to execute at improved prices
3. Market makers will be able to provide greater liquidity for odd-lot orders.

In Ethereum, PFOF has emerged as Exclusive Order Flow ("EOF") relationships between Order Flow Originators ("OFOs") and Builders. EOF bypasses the public mempool and accounts for as much as 35% of the market. Exclusive Order Flow enables a builder to construct a higher value block than competitors constrained to sourcing transactions from the public mempool or Order Flow Auctions ("OFA"). Because EOFs require execution guarantees, builders will multiplex the OFO's bundle to guarantee timely inclusion.

There are several reasons why Order Flow Originators utilize EOF relationships:

- Block inclusion guarantees. By partnering with a top builder, OFOs increase the probability that their transactions are included in the next block built.
- User MEV protection. OFOs can virtually eliminate their transactions' MEV.
- Priority gas fee rebates. OFOs can receive a refund of the priority gas fees paid by their users. Builders value high-quality transactions like sniping orders from Telegram bots and will pay more for this order flow.

Currently known EOF relationships:

- Banana Gun and Titan Builder
- Maestro and Beaver

*EOF is approximated by order flow not seen by Flashbots or in the mempool.

Order Flow Auctions

Order Flow Auctions (OFAs) were created to protect user transactions from negative MEV strategies such as front-running and sandwich attacks. OFAs offer many benefits to users including:

1. Lower transaction costs. OFAs bundle transactions which reduce gas fees and reduce execution slippage.
2. MEV refunds. OFAs can auction MEV back-run opportunities and return a portion of the captured MEV to users.
3. Improved price discovery. Third-party solvers compete for the best execution price.
4. Enhanced liquidity. Third-party solvers can aggregate liquidity from numerous sources including DEXs, CEXs, and private inventory.

OFAs aggregate swap transactions from multiple users and auction them to third-party bidders for execution. OFAs function as the auctioneers and select winning bids on predefined

criteria. The winning bids are submitted on-chain in a bundle to block builders for consensus.

There are different types of OFAs:

- Request for Quote (RFQs):
RFQs utilize a system of pre-selected bidders, funds, and market-makers, that use on-chain and private inventory to submit bids. RFQs offer better liquidity than public automated market makers (AMMs) as RFQ market makers have access to additional sources of liquidity such as CEXs and cross-chain AMMs. Example: UniswapX, Bebop, 1inch Fusion, Hashflow, 0xAPI
- Frequent Batch Auctions:
Frequent Batch Auctions enable third-party solvers to optimize for price and liquidity while protecting transactions from MEV. Transactions are bundled saving on gas and improving execution. Example: CoWSwap, DFlow
- Transaction Execution Auctions:
Third-party bidders, specifically searchers, extract MEV and compete for the highest user refund. This OFA is typically integrated directly with wallets via an RPC. Example: MEV-Blocker, Merkle
- Block Space Aggregator Auctions: Block Space Aggregator Auctions return value to the original user through builder priority gas rebates. Builders compete to include the transaction bundle to increase the value of their block and will refund a portion of the priority gas paid by the bundle. Example: Flashbots MEV Share

Private Order Flow

Private Order Flow (POF) is the order flow from vertically integrated order flow originators (wallets, applications, solvers, searchers) and builders. This flow is typically not multiplexed and sent to a singular builder.

The top builders, Beaver Build and Rsync, are integrated with proprietary trading firms SCP and Wintermute and benefit from internal CEX-DEX order flow. [Integrated searcher-builders](#) have an advantage over normal builders since profits from their searcher can be reallocated to their builder increasing their likelihood of submitting the winning block bid. Integrated searcher-builders also benefit from latency savings when sending their transaction from the searcher to the builder. This latency savings can then be extended to the block builder auction.

(a)
<https://arxiv.org/pdf/2407.13931>.
EOF for Titan (b),
Beaverbuild (c), and Rsync
builders (d). Note that only
Rsync sees Wintermute
private order flow and only
Beaverbuild sees SCP
private order flow.

Future Implications

OFAs like Flashbots

Protect and MEV-Blocker have provided RPCs for users to integrate into their wallets. These products were primarily opt-in for individual wallet users and directly integrated into applications.

Moreover, wallets have started to capture the value of their order flow.

- Metamask Smart Transactions - Metamask Smart Transactions perform the same function as an OFA providing MEV protection, gas refunds, and revert protection. This service is automatically integrated into Metamask's wallet and is opt-in for users. Searchers and solvers pay for access to Smart Transactions order flow.
- Trust Wallet MEV Protection - Trust Wallet MEV Protection is included by default for users but does not include gas refunds and revert protection.

References

1. CNBC 2021, [Virtu Financial CEO weighs in on payment for order flow crackdown](#).
2. Bradford Levy, [Wharton Initiative on Policy and Regulation - Research Spotlight: Payment of Order Flow and Price Improvement](#).
3. Thomas Thiery, [Empirical analysis of Builders' Behavioral Profiles \(BBPs\)](#).
4. CoWProtocol, [Understanding Order Flow Auctions](#).
5. Blocknative, [Exploring Order Flow Auctions \(OFAs\) - MEV and the Fair Distribution of Economic Opportunities](#).
6. Darren Kleine, [It's all about the swaps. Why order flow auctions make DEXs better](#).
7. S. Yang, K. Nayak, F. Zhang, 2024. [Decentralization of Ethereum's Builder Market](#).
8. Tivas Gupta & Mallesh M Pai & Max Resnick, 2023. [The Centralizing Effects of Private Order Flow on Proposer-Builder Separation](#).
9. Quintus Kilbourn, [Order flow, auctions, and centralisation I -](#)

- [a warning.](#)
10. Quintus Kilbourn, [Order flow, auctions, and centralisation II - order flow auctions.](#)
 11. B. Bachu, X. Wan, C. Moallemi, 2024. [Quantifying Price Improvement in Order Flow Auctions.](#)
 12. Frontier Research and Titan, 2023. [Builder Dominance and Searcher Dependence.](#)
 13. Frontier Research, [The Orderflow Auction Design Space.](#)
 14. Pai, M. and Resnick, M., [Structural Advantages for Integrated Builders in MEV-Boost.](#)
 15. Oz, B., Sui, D., Thiery, T., and Matthes, F., 2024. [Who Wins Ethereum Block Building Auctions and Why?](#)

Account Abstraction

The implementation of Account Abstraction, particularly through ERC-4337, is revolutionizing user interactions with blockchain networks by introducing smart contract wallets and new entities like Bundlers and Paymasters. These developments are not only enhancing user experience but also creating new opportunities and challenges in transaction processing and fee structures.

ERC-4337

The key goals of account abstraction are to remove the need for all users to have an EOA and to allow users to use smart contract wallets as their primary account. Account abstraction accomplishes this by separating account management and transaction execution from EOAs. Account abstraction uses new entities: 1) the Bundler, to initiate transactions and 2) the Paymaster, to determine the gas payment policies.

[ERC-4337](#)

introduces two new parties - the Bundler and the Paymaster:

- Bundler - The Bundler assembles multiple user operations into a transaction, similar to a block builder, and

submits the transaction to the entry point contract for execution. More importantly, Bundlers have EOAs that allow them to initiate transactions abstracting away the need for users to have an EOA wallet. Current bundlers include Skandha, Alchemy, Rundler, Voltaire, Alto, Stackup, and Infinitism.

- Paymaster - The Paymaster is a smart contract that handles the wallet's gas payment policies. The Paymaster determines which currency, stablecoins or other ERC-20 tokens, are acceptable for gas payments and allows applications to pay gas fees for their users.

Future Implications

Under ERC-4337, the Bundler is in a similar position to today's block builder and can

execute
exclusive
order
flow
deals
with
smart
contract
wallets.
Exclusive
order
flow
is
more
important
to
Bundlers
because
they
compete
for
the
highest
priority
fee
and
losing
Bundlers
pay
for
the
gas
cost
of
reverting
UserOperation.

Because
the
UserOperation
mempool
is
public,
UserOperations
are
susceptible
to
MEV
from
front-
running
and
sandwich
attacks.
Bundlers
can
capture
a
portion
of
this
MEV
since
they
order
and
batch
the
UserOperations
into
a
bundle
transaction.
Searchers
could
run
Bundlers
to
extract
MEV
from
the
public
UserOperation
mempool.
Bundlers
and
Builders
could
integrate

to
obtain
additional
order
flow.

References

1. [ERC-4337: Account Abstraction via Entry Point Contract Specification](#)
2. [ERC-4337: Account Abstraction Using Alt Mempool](#)
3. <https://github.com/ethereum/EIPs/pull/4337/files>
4. Alchemy, [What is Account Abstraction \(ERC-4337\)?](#)
5. Alchemy, [How do ERC-4337 smart contract wallets work?](#)
6. Blocknative, [Introductory Guide to Account Abstraction \(ERC-4337\)](#)
7. Blocknative, [Understanding ERC-4337 - How it works and exploring unknowns](#)
8. Dmarz, [4337 MEV supply chain](#)
9. Blockbase, [Does ERC-4337 reduce the impact of MEV-Boost on Ethereum?](#)
10. BlockPI Network, [Why Must ERC-4337 Bundlers Work with Block Builders?](#)
11. [Kernel](#)

[Ventures:
Understanding
Ethereum's
ERC4337
Standard](#)
-
[What
Opportunities
Lie
Ahead](#)

12. <https://www.bundlebear.com/overview>

Future Trends

The cryptocurrency and blockchain landscape is on the cusp of significant transformation, driven by technological innovations and regulatory developments. Key trends shaping the future include:

- Preconfirmations for faster transaction speeds.
- EIP-7702 and EIP-7212 for account abstraction and smart wallet improvements that enhance user experiences and transaction signing standards.
- Integration of Trusted Execution Environments (TEEs) for heightened security.
- Regulatory discourse, particularly around stablecoins and securities, is pushing the industry to adapt within

new
legal
frameworks.

As
the
focus
shifts
from
broad
decentralization
to
nuanced
discussions
of
control
and
execution,
these
trends
collectively
promise
to
redefine
how
users
interact
with
blockchain
networks,
how
developers
build
applications,
and
how
the
ecosystem
navigates
regulatory
challenges.

Preconfirmations

Preconfirmations
("preconfs")
is
a
research
proposal
that
allows
users
to
receive
a
transaction
confirmation
before
their
transaction
is
confirmed
in
consensus.
Preconfs
aim
to
improve
the
user
experience
by
eliminating
high
network
congestion
on
Ethereum,
layer
2
rollups,
and
validiums
through
faster
confirmations.
First
introduced
by

Justin
Drake,
based
preconfs
allow
L1
proposers
to
provide
economic
guarantees
that
an
L2
user
transaction
will
be
included.

How do preconfs work?

- Ethereum
block
proposers
("preconfers")
or
a
delegate
party
issue
signed
promises
to
users
guaranteeing
that
their
transactions
will
be
included
and
executed
faster
than
expected
L1
consensus.

The
preconfirmation
landscape
is
still
in
its
early
stages
and
several
different
methodologies
have
been
proposed.
These
are
the
few
that
could
affect
order
flow
originators
the
most:

- XGA-
Style
Preconfs:
XGA-
style
preconfirmations
guarantee
(non-

positional)
bundle
inclusion
in
the
bottom
portion
of
a
block.
Filler
transactions,
transactions
that
do
not
require
immediate
execution
or
have
low
MEV,
can
be
included
in
the
bottom-
of-
the-
block
bundle.
This
allows
builders
to
focus
on
valuable
top-
of-
the-
block
MEV
transactions
and
simplifies
gas
pricing
for
filler
transactions.

- MEV-Commit
by
Primev:
MEV-
commit
is
a
P2P
network
where
execution
commitments
for
Ethereum
transactions
are
committed
and
providers
are
rewarded
or
slashed.
Order
flow
originators
("bidders")
specify
their
intents
for
transaction
executions
to
providers.

- BFT
Preconfirmations
by
Espresso:
BFT
preconfirmations
are
backed
by
security
and
liveliness
guarantees
of
a
BFT
consensus
algorithm.
BFT
preconfs
are
backed
by
a
subset
of
L1
validators
and
not
a
single
validator
like
in
based
preconfs.

Future Implications

Preconfs
will
lead
to
a
better
execution
experience
since
order
flow
originators
can
guarantee
transaction
execution
for
higher
fees.

In
the
case
of
XGA-
style
preconfs,
bottom-
of-
the-
block
inclusion
for
non-
latency-
sensitive
transactions
(i.e.
"governance",
"staking",
"authorizations",
"claiming")
can
lower
the
gas
spent
on

these
transactions
and
reduce
the
number
of
transaction
reverts
from
insufficient
gas.

References

1. Justin Drake, [Based Confirmations.](#)
2. Raghav Agarwal, [Preconfirmations: Credible Promise of Future Execution](#)
3. Murat Akdeniz - Primev, [Preconfirmations: The Fulfillment-Delivery Paradigm.](#)
4. CTra1n, [Value-Capturing Based Rollups with Based Preconfirmations.](#)
5. dpl0a, [Preconfirmations: On splitting the block, mev-boost compatibility and relays.](#)
6. Ellie Davidson - Espresso Systems, [Analyzing BFT and Proposer-Promised Preconfirmations.](#)
7. Nethermind, [RFP-001: Introduce preconfirmation infrastructure.](#)
8. Cairo, [Towards an Implementation of based preconfirms.](#)

**EIP-7702
and
EIP-7212**

There
are

two
Account
Abstraction
EIPs
that
could
fully
unlock
the
potential
of
smart
contract
wallets
and
become
game-
changers
for
the
wallet
ecosystem.

EIP- 7702

EIP-
7702
introduces
the
following
features
to
EOAs:

- **Batching:**
A
user
can
perform
multiple
operations
in
one
atomic
transaction.
- **Sponsorship:**
A
separate
account
X
or
application
operator
can
pay
for
account
Y's
transaction.
Account
X
can
receive
ERC-
20
tokens
for
this
service.
- **Privilege
de-
escalation:**
Users
sign
sub-
keys
that
provide
weaker,
specific
permissions.
For
example,
interacting
only
with
specific
applications,

using only certain ERC-20 tokens for a transaction, and transfer limits.

EIP-7702 is designed to be backward and forward compatible with ERC-4337 allowing EOAs to take advantage of the existing ERC-4337 infrastructure. EOAs can also temporarily convert themselves into smart contract wallets for inclusion in ERC-4337 bundles.

Benefits of EIP-7702 include:

- Less security risk: EIP-7702 also eliminates the central point of trust when assigning smart contract codes to EOAs for a transaction. There is no possibility of unauthorized transactions with

- EIP-7702 since the contract code is removed after the transaction is executed.
- Easy adoption for dApps: Applications using ERC-4337 can easily integrate with EIP-7702 without any changes to their code. EOAs can call the smart contract without any need for authorization.

EIP-7702 is still a new proposal and has a few issues that developers need to consider:

- Revocations: EIP-7702 does not have clear details on revoking contract code in case any malicious code is detected.
- Chain Agnostic Signatures: EIP-7702 uses a fixed

signature
that
can
be
reusable
in
other
chains
but
lacks
flexibility
if
users
want
different
implementations.

EIP-7212

EIP-7212,
or
[RIP-7212](#),
creates
a
contract
for
signature
verification
using
the
"secp256r1"
elliptic
curve
standard.
This
standard
has
been
adopted
for
user
authentication
by
the
largest
Web2
corporations
and
can
be
integrated
into
ERC-4337's
smart
contract
wallets.

"secp256r1"
is
currently
used
in
the
following
authentication
applications:

1. Apple's
Secure
Enclave:
Apple's
Secure
Enclave
is
Trusted
Execution
Environment
(TEE)
hardware
that
creates
and
stores
keys.
The
Secure

- Enclave
can
encrypt
or
decrypt
data,
sign
arbitrary
messages,
and
is
only
accessible
through
biometric
identification.
2. WebAuthn:
Web
Authentication
is
a
web
standard
for
authentication
used
by
most
Web2
browsers
-
Chrome,
Firefox,
Edge,
and
Safari.
WebAuthn
uses
domain-
specific
public-
key
cryptography
for
user
authentication
eliminating
passwords,
providing
faster
recovery,
and
reducing
security
risks.
3. Android
Keystore:
Android
Keystore
is
a
secure
system
credential
storage.
Applications
can
create
private
and
public
keys
and
store
them
in
the
Keystore.
The
Keystore
is
encrypted
based
on
the
user's
own
mobile
password
and

can
be
accessed
via
password
or
biometrics.
4. Passkeys:
Passkeys
are
FIDO
credentials
that
allow
users
to
access
their
accounts
without
passwords
using
biometrics
or
a
PIN.
Users
can
access
websites
or
apps
by
unlocking
their
mobile
devices
eliminating
the
need
for
passwords.

RIP-
7212
is
the
roll-
up
version
of
EIP-
7212
and
teams
from
Kakarot,
Polygon,
Optimism,
zkSync,
Scroll,
and
Arbitrum
have
already
committed
to
implementation.
Polygon
has
RIP-
7212
available
on
their
testnet
and
Coinbase's
recently
launched
Smart
Wallets
include
passkey
authentication.

**How
do
passkeys**

work?

1. A smart contract wallet creates a passkey or public and private key pair.
2. The private key is stored in a TEE on your mobile device.
3. When the smart contract wallet creates a transaction for approval, the user authenticates themselves with biometrics or a mobile PIN to unlock the private key.
4. The mobile device then uses the private key to "sign" the transaction and sends the completed transaction back to the smart contract wallet.
5. The signature is verified on-chain through the RIP-7212 smart contract.

Future

Implications

While EIP-7702 is still a proposal, RIP-7212 is being actively integrated into L2 roll-ups and implemented into smart contract wallets. Passkey wallets supercharge ERC-4337 smart wallets by eliminating the need for passwords and seed phrases and elevating security to a hardware level. Current projects featuring passkeys include:

- **Coinbase Smart Wallet:** Coinbase's Smart Wallet utilizes a passkey for user authentication and sponsored gas transactions. Smart wallets support 8 networks (Base, Ethereum, Optimism, Arbitrum, Polygon, Avalanche, BNB, Zora) and offer a wallet SDK for dApps

integration.

- **Clave:**

Clave
utilizes
mobile
TEEs
and
passkeys
to
offer
social
recovery,
account
naming
services,
biometric
login,
and
sponsored
gas
fees
on
zkSync.

- **Banana**

SDK:
Banana's
SDK
utilizes
WebAuthn
to
offer
zero-
knowledge
2FA,
biometrics,
and
recovery
accounts
with
a
nominee.

- **Turnkey:**

Turnkey
is
a
Wallet-
as-
a-
Service
(WaaS)
provider
that
stores
private
keys
in
a
TEE.

References

1. Coinbase,
[How
Base
is
making
smart
wallets
the
default](#)
2. Github,
[EIP-
7702](#)
3. Zyfi,
[Into
the
future
with
EIP-
7702
-
Part-
1](#)
4. Quicknode,
[EIP-
7702
Explained:
The
Future](#)

- [of
Ethereum](#)
5. [Web3Auth,
EIP-
7702
Explained:
How
it
Works
and
Everything
You
Need
to
Know](#)
 6. [Erdogan,
U.,
Alpaslan,
D.,
Posch,
DC.,
Bhardwaj,
N.,
EIP-
7212:
Precompile
for
secp256r1
Curve
Support.](#)
 7. [Alchemy,
What
is
RIP-
7212?
Precompile
for
secp256r1
Curve
Support.](#)
 8. [Apple,
Secure
Enclave.](#)
 9. [WebAuthn,
WebAuthn
Guide.](#)
 10. [Android
Authority,
How
to
use
the
Android
Keystore
to
store
passwords
and
other
sensitive
information.](#)
 11. [Google,
Ask
a
Techspert:
What
are
Passkeys?](#)
 12. [Coinbase,
Smart
Wallet
Documentation](#)
 13. [Clave
FAQ](#)
 14. [Banana
Wallet
SDK
Docs](#)

EIP- 712

EIP-
712
is
a
standard
for
typed

message
signing
which
aims
to
allow
off-
chain
message
signing
for
on-
chain
signing
allowing
for
a
better
user
experience.
Rather
than
reading
byte
strings,
EIP-
712
enables
signatures
to
be
displayed
in
a
readable
format
without
losing
system
security
properties.
Off-
chain
signing
saves
gas
and
reduces
the
number
of
transactions
on-
chain.

How does EIP- 712 work?

1. dApps
developers
utilize
a
JSON
data
structure
that
users
sign.
2. A
domain
separator
prevents
the
signature
from
being
used
on
multiple
dApps
and
allows
for
multiple
distinct
signature

use cases within a given dApp.
3. Wallets and front-end operators can parse the dApp data structure and translate the data into a readable message for users.

One of the key features that EIP-712 unlocks is that it allows dApps to control the transaction flow for users rather than wallets. Applications like Uniswap, can minimize their users' MEV since swaps would bypass OFAs and other MEV value extractors.

In addition to wallet transaction readability, EIP-712 improves governance usability by allowing a third party to pay the

gas
fees
for
user
votes.
Voters
can
use
EIP-
712's
by-
signature
functionality
to
create
a
signed
delegate
or
vote
transaction
for
free.

Future Implications

In
addition
to
wallet
readability,
EIP-
712
can
be
used
to
improve
the
user
experience
in
other
areas.

- Governance.
Users
can
delegate
their
vote
and
have
a
third
party
pay
the
gas
fees
for
them
through
EIP-
712's
by-
signature
function.
- Clear
Signing.
Hardware
wallets,
or
separate
devices,
can
display
the
dApp's
message
ensuring
that
users
can
be
certain
that
no

malware
or
malicious
application
has
sent
the
message
to
them.

- Replay
attack
prevention.
The
data
to
prevent
replay
attacks
can
be
included
inside
the
EIP-
712's
structured
data.
- MEV
minimization.
EIP-
712
allows
a
user
to
sign
a
transaction
while
giving
front-
ends
the
ability
to
send
the
order
flow
without
broadcasting
it
to
the
network
minimizing
a
user's
exposure
to
malicious
MEV.

References

1. [EIP
712.
Typed
structured
data
hashing
and
signing](#)
2. Koh
Wei
Jie,
[EIP712
is
here:
What
to
expect
and
how
to
use
it](#)
3. Adam

Bavosa,
[Delegation
and
Voting
with
EIP-
712
Signatures](#)

4. SpruceID,
[Sign-
In
with
Ethereum
Wallet
Research](#)
5. Ledger
Tech,
[Messages,
Transactions,
and
Clear-
Signing](#)
6. Cyfrin,
[EIP712
and
EIP191](#)
7. Metamask
Github,
[Sign
transaction
without
broadcast
#3475](#)

Trusted Execution Environments (TEEs)

Trusted
Execution
Environments
(TEEs)
is
a
secure
enclave
based
within
a
hardware
microprocessor
where
sensitive
computations
and
operations
can
run
with
integrity
and
privacy.
TEEs
support
isolation
and
remote
attestation
and
can
run
virtual
machines
like
EVM
and
CosmWasm
without
the
cryptographic
overhead
like
Multi-
Party
Computation
(MPC)
or
zkSNARKs.

For
web3
wallets,
mobile
TEEs
like
Apple's
Secure
Enclave
and
Google's
Titan
M2
can
secure
smart
contract
wallet's
private
keys
better
than
standard
hardware
wallets.
Users
can
create
and
store
a
private
key
inside
a
TEE
and
sign
transactions
from
these
keys.
The
keys
remain
on
the
device
and
can
only
be
accessed
by
the
device
owner
via
biometric
authentication
or
device
PIN.

TEEs
are
currently
used
in
several
wallet
solutions:

- MPCs:
Fireblocks
utilizes
Intel
SGX
TEEs
to
isolate
cryptographic
data,
the
MPC
and
ZKP
cryptographic
algorithms,
and

the execution portions of their software from their internal systems and external third parties. Fireblocks stores MPC keys, API credentials, and their Policy Engine in the secure enclave to prevent unauthorized access by hackers, rogue employees, and inside colluders.

- Smart Contract Wallets: As listed in the previous RIP-7212 section, smart wallets leverage mobile TEEs to store passkeys. Current smart contract wallets using TEEs include Coinbase Smart Wallet, Banana SDK, Turnkey, Clave, and Weeve.

Future Implications

TEEs are poised to be a major game changer for blockchains.

- Flashbots
SUAVE
will
utilize
TEEs
to
create
a
secure
and
private
MEV
ecosystem.
- Smart
contract
wallets
will
use
mobile
TEEs
and
Account
Abstraction
to
improve
the
user
onboarding
experience
and
reach
a
new
user
audience.
- Large
corporations
have
adapted
TEEs
to
solve
their
own
privacy
and
security
needs.
Visa
created
the
LucidiTEE
blockchain
that
improves
multiparty
computation
and
storage
for
private
data.
- TEEs
are
a
potential
regulatory
compliance
solution
to
enhancing
control,
data
privacy,
and
operational
security
for
blockchains.

References:

1. Andrew
Miller,
[TEE-
based
Smart
Contracts
and
Sealing](#)

- [Pitfalls.](#)
2. Braavos,
[Hardware
Signer:
Enhancing
Security
of
Crypto
Wallets.](#)
3. Base,
[Solidity
WebAuthn
Authentication
Assertion
Verifier.](#)
4. Nick
Summers,
[Passkeys
FAQs:
What
they
are,
and
other
frequently
asked
questions](#)
5. Apple,
[Secure
Enclave
Documentation](#)
6. Calvin
Wankhede,
[What
is
the
Titan
M2
security
chip
in
Google's
Pixel
phones?](#)
7. [Fireblocks
Documentation](#)
8. Sinha,
R.,
Gaddam,
S.,
Kumaresan,
R.
-
Visa
Research.
[LucidiTEE:
A
TEE-
Blockchain
System
for
Policy-
Compliant
Multiparty
Computation
with
Fairness.](#)

Execution, Control, & Decentralization

One
of
the
major
challenges
for
wallet
providers
has
been
educating
users
and
regulators

about
self-
custody
and
on-
chain
accounts.
However,
as
the
web3
ecosystem
has
matured,
key
stakeholders
have
come
to
understand
several
crucial
points:

1. Wallets
do
not
hold,
manage,
or
custody
user
assets;
they
merely
provide
an
interface
to
access
them.
2. Blockchain
addresses
exist
independently
of
wallets
and
are
not
created
or
managed
by
them.
3. Users
have
the
freedom
to
switch
between
different
wallet
providers,
as
their
assets
and
accounts
exist
on-
chain,
separate
from
any
specific
wallet
service.

This
growing
understanding
has
been
crucial
in
clarifying
the
role

of
wallets
in
the
cryptocurrency
ecosystem
and
distinguishing
them
from
traditional
financial
service
providers.

Stablecoins and MiCA

Stablecoins
continue
to
be
among
the
most
significant
crypto
assets
as
they
enable
seamless
and
frictionless
transfer
of
value
across
borders
and
economic
systems.
They
permit
users
to
move
value
between
assets
that
may
fluctuate
in
price
to
stable
denominations
for
future
use.
However,
stablecoins
have
risen
to
the
top
of
many
regulators'
crypto
agendas
primarily
due
to
concerns
about:

1. Controlling
and
monitoring
the
flow
of
global
fiat
currency

flows.

2. Their influence on currency strength and monetary policy.
3. The need for oversight in their issuance and backing.

As a result, stablecoins have become a top priority on many regulators' cryptocurrency agendas, sparking debates about their role in the broader financial ecosystem.

In the United States, stablecoins gained significant regulatory attention with Facebook's (now Meta) Libra project. The tech giant proposed a privately managed stablecoin that could, in theory, become the predominant digital currency, raising concerns about its impact on central banks' monetary policy control.

Since the

Libra project shuttered in 2022, the primary goal of the US regulatory stablecoin policy has been ensuring the proper collateralization and oversight of stablecoins. This shift has led stablecoin issuers to adopt practices similar to regulated financial institutions with robust custody agreements, established banking relationships, and comprehensive monitoring programs. While various regulatory agencies have contributed piecemeal regulations, the US Congress is working towards a more comprehensive regulatory framework for stablecoins.

In the EU, the Market in Crypto-Assets Regulation (MiCA) is rolling into effect and contains key stablecoins provisions.

As
of
this
writing,
only
Circle's
USDC
and
Euro
stablecoin
have
successfully
registered
in
the
EU.

Swaps and Securities Regulation

In-
wallet
token
swaps
have
become
a
popular
feature
in
many
cryptocurrency
wallets
improving
the
usability
of
on-
chain
applications
and
enabling
users
to
navigate
bridging
and
cross-
chain
interactions.

However,
this
functionality
has
attracted
regulatory
scrutiny,
particularly
from
securities
regulators,
attempting
to
apply
traditional
financial
services
regulations
to
wallets
offering
swap
features.
Most
notably,
the
SEC
has
taken
legal
action
against
certain
wallet
providers
alleging

that
these
wallets'
swap
functions
effectively
operate
as
unregistered
broker-
dealers.

**SEC
v.
Coinbase**

In
April
2024,
the
SEC's
[claim](#)
that
Coinbase
Wallet
acted
as
a
broker
was
dismissed.
Self-
custody
wallets
with
swap
functionalities
generally
do
not
meet
the
criteria
for
broker
classification.
The
SEC's
argument
is
based
on
their
allegations
that
some
assets
available
through
these
wallets
are
unregistered
securities..

**SEC
v.
Consensys**

Consensys
proactively
sued
the
SEC
in
April
2024
over
whether
the
SEC
has
the
legal
authority
to
regulate
MetaMask
as

a
securities
broker
and
issuer
and
was
granted
an
expedited
review
by
the
judge
in
the
case.
The
expedited
court
proceedings
could
lead
to
a
decision
by
the
end
of
this
year.

Despite
this
lawsuit
and
losing
the
Coinbase
v
SEC
lawsuit,
the
SEC
filed
a
[Wells
Notice](#)
against
Consensys
at
the
end
of
June
2024.
The
SEC
alleged
that
Consensys
acted
as
an
unregistered
broker
of
crypto
asset
securities
through
MetaMask
Swaps
and
through
its
crypto
staking
program,
MetaMask
Staking.

Future Implications

While
wallets
will

continue
to
be
at
the
forefront
of
debates
over
illicit
finance
and
self-
custody,
much
of
the
future
regulatory
conversation
will
pivot
to
the
question
of
decentralization.
For
the
past
few
years,
the
crypto
industry
has
leveraged
the
concept
of
decentralization
to
explain
to
regulators
why
traditional
financial
securities
regulations
should
not
apply
to
crypto
services.
This
argument
specifically
addresses
the
questions
of
control
and
responsible
parties.

Traditional
finance
rules
and
guidance
regulate
intermediaries
to
provide
consumer
protection
and
accountability.
However,
a
key
challenge
emerges:
how
do
you
achieve

these
objectives
when
the
services
involved
are
inherently
not
intermediaries
and
do
not
custody
assets
or
execute
operations
for
users?

Decentralization,
both
as
a
concept
and
a
design
goal,
has
helped
explain
why
traditional
financial
services
regulations
are
difficult
to
apply
to
crypto.
However,
we
are
now
entering
a
new
phase
of
regulatory
discourse
where
regulators
are
seeking
to
define
and
apply
definitions
of
decentralization
to
various
services,
from
wallets
to
decentralized
exchanges
(DEXs)
and
beyond.
Regulators
now
see
an
opportunity
to
classify
many
crypto
services
as
non-
decentralized

or
"decentralized-
in-
name-
only."
This
classification
stems
from
two
main
factors:

1. The
burden
of
meeting
standards
for
true
decentralization
is
often
technically
unfeasible
for
many
services.
2. These
standards
of
decentralization
may
not
align
with
the
actual
goals
of
regulation.
This
shift
in
regulatory
approach
could
have
significant
implications
for
how
crypto
services
are
classified
and
regulated
in
the
future.

That
is
why
the
next
phase
of
regulatory
discourse
will
shift
to
the
concept
of
control.
Key
questions
will
include:
Do
wallets
have
control
over
the
execution
of

a
user's
operation?
Do
DEXs
have
control
over
how
an
operation
is
executed
or
filled?
The
crypto
industry
as
a
whole
is
making
significant
progress
in
developing
new
operational
models
that
move
beyond
the
notion
of
decentralized
services
and
into
a
conversation
about
control,
data,
and
privacy.

At
the
forefront
of
these
advancements
is
the
utility
of
trusted
execution
environments
(TEEs).
We
are
moving
towards
a
market
structure
where
operational
control
resides
within
hardware
and
software,
rather
than
with
service
providers.
In
this
model,
service
providers
do
not
have

direct
control
over
the
operations
taking
place
nor
the
ability
to
view
user
orders.
With
this
approach,
the
crypto
industry
is
pioneering
novel
ways
for
financial
services
and
communications
applications
to
operate..

Lastly,
as
we
shift
from
discussions
about
decentralization
to
more
nuanced
conversations
about
control,
the
concepts
of
execution,
finality,
and
settlement
will
become
increasingly
important.
The
industry
will
need
to
define
collectively:

1. Who
is
responsible
for
executing
an
operation
2. When
an
operation
is
considered
settled
on-
chain
3. Who
is
responsible
for
its
settlement

Contributors

Flashbots

Mates:

Tesa

Ho,

George

Zhang,

Reid

Yager,

Quintus

Kilbourn,

Fred,

Danning

Sui,

Elaine

Hu,

Daniel

Marzec.

Friends:

Ivo

Georgiev,

Joe

Huang,

Nic

Lin,

Tina

He,

Nico

Csgy,

Eric

Siu,

Samuel

Akpan,

Brian

Friel,

Andre

Geest,

Daniel.

```
{[
```

```
{
```

```
src:
```

```
"/img/state_of_wallets_figures/ambire_wallet_logo.png",
```

```
alt:
```

```
"Ambire
```

```
Wallet
```

```
logo",
```

```
caption:
```

```
"Ambire
```

```
Wallet",
```

```
},
```

```
{
```

```
src:
```

```
"/img/state_of_wallets_figures/okx_wallet_logo.png",
```

```
alt:
```

```
"OKX
```

```
Wallet
```

```
logo",
```

```
caption:
```

```
"OKX
```

```
Wallet",
```

```
},
```

```
{
```

```
src:
```

```
"/img/state_of_wallets_figures/rainbow_wallet_logo.svg",
```

```
alt:
```

```
"Rainbow
```

```
Wallet
```

```
logo",
```

```
caption:
```

```
"Rainbow
```

```
Wallet",
```

```
},
```

```
{
```

```
src:
```

```
"/img/state_of_wallets_figures/imtoken_wallet_logo.png",
```

```
alt:
```

```
"imToken
```

```
Wallet
```

```
logo",
```

```
caption:
```

```
"imToken
```

```
Wallet",
```

```
},
```

```
{
```

```
src:
```

```
"/img/state_of_wallets_figures/base_logo.png",
```

```
alt:
"Base
Wallet
logo",
caption:
"Base",
},
{
src:
"/img/state_of_wallets_figures/cw_logo.png",
alt:
"CW
logo",
caption:
"CW",
},
{
src:
"/img/state_of_wallets_figures/safe_wallet_logo.png",
alt:
"Safe
Wallet
logo",
caption:
"Safe
Wallet",
},
{
src:
"/img/state_of_wallets_figures/block_daemon_logo.png",
alt:
"Blockdaemon
logo",
caption:
"Blockdaemon",
},
].map((logo,
index)
=>
(

{logo.caption}
))}
```