CoinEx Chain

白 皮 书

CoinEx

目录

概要

数字货币的中心化交易所全权掌控了用户的资产，而频发的中心化交易所被盗事件一次次表明其安全问题成为悬在整个数字货币行业之上的达摩克利斯之剑。2011年和2014年Mt.Gox两次被盗，2014年Poloniex被盗，2015年Bitstamp被盗，2017年Bithumb被盗，2019年Binance被盗等事件一次次印证了前述的担忧。在资产安全的问题之外，交易所关门跑路、交易规则不透明、由于主观或客观原因造成的暂时无法访问以及昂贵的上币费用等也是中心化交易所常被诟病的地方。

能否以去中心化的方式重构数字货币的交易市场？不需要注册要求或者审核流程，没有单点故障问题、控制或者监管属性，规则透明的去中心化交易所能够同时解决中心化交易所面临的诸多挑战。近年来涌现出诸如Bitshares [1]、Etherdelta [2]、0x协议 [3]、OmiseGo [4]、Loopring [5]、 Kyber [6]、Cosmos[7] 等去中心化交易所的技术方案。现有的大部分方案，如Etherdelta、0x协议、OmiseGo、Loopring、Kyber都是基于以太坊的支持ERC20标准的去中心化交易解决方案。基于某个公链的去中心化交易平台的能力则受限于底层公链的能力。以以太坊为例，在当下以太坊的交易处理能力迟迟无法提升的情况下，基于以太坊构建的去中心化交易平台在处理速度、用户体验等方面无法与现有的中心化交易平台匹敌。专门定制的ASIC芯片已经在基于PoW的公链挖矿中展现出巨大的优势，借鉴同样的思路，构建专门用于去中心化交易的公链能够在规避中心化交易所诸多问题的同时，依然保持较高的交易处理速度和相似的用户交互体验。

CoinEx Chain是基于Tendermint共识协议[7,8]和Cosmos SDK[9,10]打造的公链，旨在打造社区化运营、交易规则透明以及用户资产由自己控制的去中心化交易平台（DEX）。 基于CoinEx Chain的DEX上的原生代币为CoinEx Token（CET），现存的ERC20 Token形态的代币会按照1:1的比例映射成CoinEx Chain上的原生代币。

Tendermint共识协议在保证足够去中心化的前提下能到达到上万的TPS以及秒级确认，能够在去中心化交易平台场景下提供近乎中心化交易平台的用户体验。通过链上交易、链上撮合的方式带来最大程度的交易透明度。将数字资产的控制权还给用户并且通过数字签名进行权限鉴定，避免了中心化交易平台所带来的单点失败的安全风险，保证了用户资产的安全性。另外通过跨链机制，可以将当前的CET Token连接至更为广阔的数字货币世界。

CoinEx Chain不仅是一条专用于DEX的公链，而且是围绕DEX公链构建的更为丰富的生态系统。 为最大程度提升DEX公链的交易处理速度，DEX公链仅支持必要的功能而不支持智能合约。但是智能合约功能是构建更为复杂的金融应用的基础，因此在DEX公链之外CoinEx Chain将包含一条支持智能合约功能的Smart公链。通过跨链机制连通Smart公链与DEX公链，在保证DEX公链性能的同时，也支持复杂的金融应用。

当前区块链上的隐私问题备受诟病，因为地址概念提供的匿名特性容易通过数据进行逆向分析，最大程度地保护用户隐私是CoinEx Chain的核心任务之一。与Smart公链类似，通过构建专门的支持交易隐私保护的Privacy公链并进行跨链连通，能够提升整个CoinEx Chain生态的隐私特性。

在三条各司其职的公链之外，CoinEx Chain将在多个技术方向进行技术创新与改进：

1）安全方面，还权于用户之后如何保证用户账户私钥的安全性成为核心问题。在钱包的私钥保护方面提供阈值多方ECDSA签名机制[12,13]。与Shamir秘密分享方案 [11]（Shamir’s Secret Sharing，SSS）相比，阈值多方ECDSA签名机制在签名时，直接利用各个私钥的分片进行计算即可得到最终的签名值，无需重构原始私钥信息，能够规避SSS方案的单点失败的隐患。

2）共识协议方面，Tendermint协议需要验证者集合对每个提议的区块进行投票（签名），每个区块中的投票信息会随着验证者集合的增大而线性增长，消耗可观的链上存储空间，聚合签名可解决这一问题。聚合签名所面临的Rogue Public-Key攻击可以通过与共识场景结合进行规避，或者采用在Plain Public Key模型下安全的聚合签名机制，如Maxwell等人提出的MuSig [14] 以及Boneh等人提出的BLS聚合签名机制 [15]。

3）执行效率方面，参考以太坊的经验，区块链所采用的可认证数据结构（Authenticated Data Structure，ADS）的效率会显著影响链上交易的处理速度，Cosmos项目基于IAVL+数据结构构造了自己的ADS方案[16]，在效率方面较以太坊提出的Merkle Patracia Tree（MPT）[17] 没有明显改进。通过工程手段可以部分缓解可能由ADS结构带来的读写性能瓶颈问题。CoinEx Chain也会关注ADS设计方面的最新进展，尝试从ADS结构角度优化公链性能。

公链组件

Tendermint Core与Cosmos SDK

CoinEx Chain是基于Tendermint Core和Cosmos SDK构建的。Tendermint Core项目封装了P2P网络通信以及Tendermint共识协议，Cosmos SDK通过模块化方式提供了应用层开发所依赖的基础功能，两个组件之间通过区块链应用层接口（Application Blockchain Interface，ABCI）进行交互。

Tendermint Core处理交易时，不关心交易的具体内容，而是将交易看做字节数组。应用层根据打包好的区块，依次解释执行各个交易并更改状态信息。

Tendermint共识协议是半同步的拜占庭共识协议，具有简洁、高效和可追责的特点。共识协议的达成是在已知的验证者集合内完成的，每个验证者通过其公钥进行鉴别。具体的共识过程通过多轮的两阶段（Prevote和Precommit）投票协议以及相应的锁定机制完成。每一轮开始时通过Round-Robin的形式选取一个验证者为区块提议者（Proposer），由该验证者打包并提议区块，随后验证者就该区块的合法性进行两阶段投票，如果每个阶段都能获得来自多于2/3的验证者的投票则该区块会被提交到链上执行。需要执行多轮的可能原因有：被选中的验证者不在线，提议的区块不合法，在某个投票阶段没有收集到超过2/3的投票信息等等。为了简化对不确定因素的处理，Tendermint中每张投票有两种用途：确认合法信息和确认无效信息。根据投票信息确认当前区块或者进入下一轮，避免了PBFT共识算法中复杂的视图转化协议。可追责的特性则由公钥可鉴别验证者这一约束提供。

由于CAP定理 [18]的客观存在，Tendermint协议在安全性与可用性之间选择了安全性。也因此Tendermint共识协议有可能会短暂停止直到超过2/3的验证者达成共识。当系统中的恶意的验证者小于1/3时，Tendermint提供了永不分叉的保证。安全性优先于可用性以及永不分叉的承诺对于金融应用至关重要。CoinEx Chain在项目启动时计划支持42个节点，根据Tendermint共识协议的实验数据，在42个节点遍布五大洲的条件下Tendermint能够达到4000TPS的处理速度，能满足去中心化交易所的需求。伴随高TPS的并不是交易确认时间的延长，Tendermint共识机制提供逐区块最终化的特性，能够在秒级完成交易确认。

Proof-of-Stake

Tendermint协议假设了验证者集合的存在，在每一轮的两阶段投票协议中以带权重的Round-Robin的形式选取当前轮的区块提议者。CoinEx Chain采用Proof-of-Stake机制，生态中任何实体都可以通过抵押CET代币的形式参与验证者的竞选。验证者集合不是固定不变的，生态参与方可以发送交易来增减自己抵押的代币数量。通过跟踪这一变化并根据更新后的抵押代币的数量状态生成新的验证者集合。

抵押与惩罚机制的引入能够规避PoS机制所面临的Nothing-At-Stake的问题 [19]。PoS机制面临的另一个重要挑战是长程攻击的问题 [20]，问题的根源在于PoS链中创建新的区块只需要足够的投票而不是像PoW机制中那样需要耗费大量资源。当攻击者设法获得的某一历史时刻的验证者的私钥数量超过2/3时，就可以从那个历史时刻进行分叉，导致新设立的节点或者长时间离线的节点无从判断哪条链是真正的主链。CoinEx Chain沿袭Cosmos Hub中的策略，通过三种措施抵御长程攻击：

1）解绑周期（Unbounding Period）：验证者取回抵押的代币时，需要经过一个解绑周期才能取回自己的代币，当前的解绑周期为3周时间；
2）弱主观性（Weak Subjective）[21]：新节点第一次连接网络时或者节点长时间下线后再次上线时需要通过可信节点验证近期的区块哈希值，CoinEx Chain基金会将提供可信节点服务；
3）按时上线同步验证者集合：在一个解绑周期的时长内节点需要同步验证者集合的更新。

这三种策略都在某种程度上缓解了长程攻击的问题，但无法从根本上解决长程攻击。可验证延迟函数（Verifiable Delay Function，VDF）的提出与研究进展为从根本上解决长程攻击提供了另外一种思路 [22, 23, 24, 25]。Increment VDF概念的提出与研究进展为抵御长程攻击提供了新的工具。CoinEx Chain团队会持续关注VDF研究领域的进展，并利用前沿进展提升PoS机制安全性。

验证者是维护公链状态一致性的关键角色，而运行全节点也需要付出成本，因此CoinEx Chain会对验证者进行奖励，奖励包含两部分：新的区块奖励以及区块中包含的交易手续费。区块链项目中新的区块奖励通常依靠铸造新币来完成：基于PoW机制的比特币通过挖矿铸币而基于PoS机制的Cosmos Hub中则依靠通胀铸币。通过铸造新币的方式会与此前承诺的CET永不通胀的理念相违背。CoinEx Chain的应对策略是从预留的代币中拿出一部分进行区块奖励，此外承诺的关于CET的回收机制依然会如期进行。验证者的恶意行为或者验证者没有能力保证验证节点的稳定性，都会影响公链状态的稳定性，此类情形下CoinEx Chain会对相应验证者进行惩罚。对于验证者在同一个区块高度对两个不同区块进行投票等直接违反共识协议安全性的行为，会扣除验证者抵押的代币中可观的一部分比例作为惩罚，同时验证者将被永久性地剔除出验证者队列；对于无法维持验证节点可靠性的行为，会扣除验证者抵押的代币中的一小部分作为惩罚警示，并被禁止在一段时间内参与验证。同时被扣除的代币将会被系统统一回收用于将来的社区激励等事项。

账户与交易

CoinEx Chain是基于账户模型的，每个账户原生支持多币种，也因此CoinEx Chain原生支持多币种的转账。为了防止链上出现大量的僵尸账户消耗链上资源，CoinEx Chain要求每个账户被激活之后才可以使用，具体的激活方法是向新账户发起CET转账交易，并从新账户

的应收CET中扣除1个CET作为账户激活的功能费。每个交易可以包含多个消息，每个消息可以完成不同的操作，比如转账、奖励提取等操作。对账户权限的鉴定通过交易的签名验证来进行。签名算法是基于secp256k1曲线的ECDSA算法。CoinEx Chain支持多签交易，目前的多签交易采用了与Bitcoin中多签类似的方式，也即把多个签名和公钥信息包含在交易内。这种方式容易实现，但也有存储计算等资源的占用和消耗问题。CoinEx Chain会针对每笔交易收取交易手续费，并且只有CET可以作为交易手续费。交易手续费包含两个部分：通常意义上的Gas费用以及功能费。Gas费根据交易的字节数、所需验证的签名个数以及对存储的读写次数和字节数等进行计费，功能费则是对某些特定的操作收取额外的费用。需要缴纳功能费的操作有：DEX公链中创建新币和创建新交易对的交易，激活新账户的交易以及带锁定功能的转账交易。另外会根据被撮合交易中的交易金额按比例收取佣金，这部分也归入功能费。

CoinEx Chain计划改进多签交易的构造方式，通过采用聚合签名算法可以对多个签名值/公钥进行压缩。这种方式能够节省链上存储空间，减少需要验证的签名的数量。对于n-of-n的多签交易能够提升多签交易的隐私属性，因为聚合的公钥和签名信息能够隐藏多签交易涉及到的实体。在支持脚本系统的区块链上，借助Merkle证明对于m-of-n的多签交易可以达到同样的隐私保护效果 [26]。然而如何在没有脚本系统的条件下，达到相同的效果仍需进一步调研。聚合签名算法所面临的最大挑战是在Rogue Public Key攻击存在的前提下在Plain Public Key Model下保证安全性。Rogue Public Key利用了这一事实，允许攻击者在其他参与方不知情的情况下炮制出合法的聚合签名。通常的应对方案有两种，要求参与方证明自己确实有相应的私钥（KOSK，Knowledge of Secret Key）或者要求在待签名消息前面级联参与方的公钥。要求KOSK证明在实际中难以操作而级联公钥的做法会部分抵消聚合签名机制的效率提升效果。Plain Public Key Model下参与者无需证明自己持有所宣称的公钥对应的私钥。Blockstream研究人员设计的MuSig多签机制 [14] 与基于双线性对的BLS签名机制 [15] 能够满足安全要求并且没有前述两种方法的弊端。

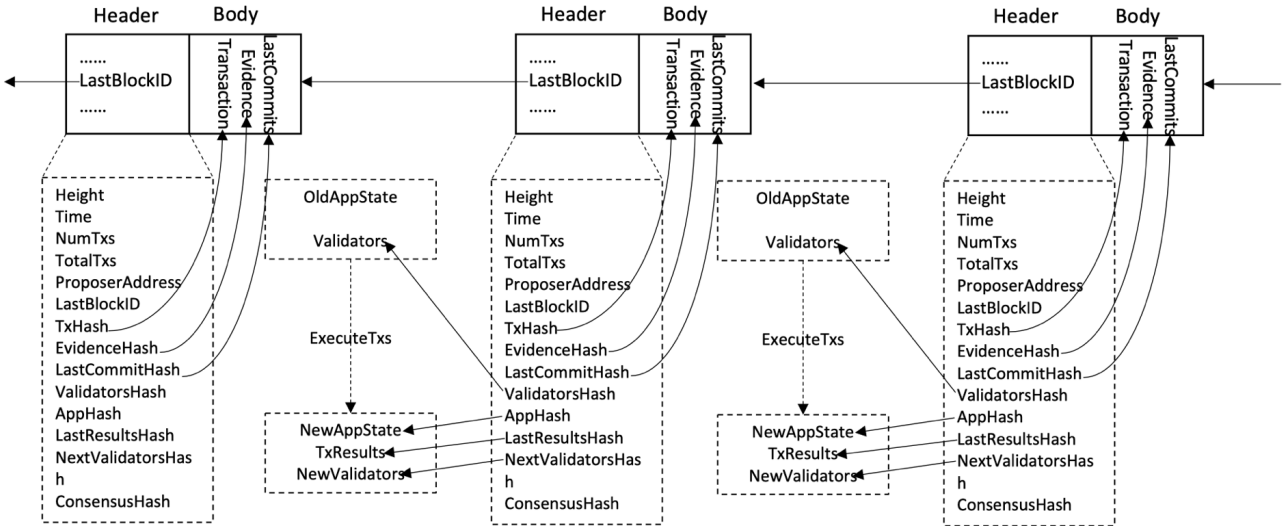
基于Schnorr签名 [27] 构建的MuSig签名方案 [14] 可以在Plain Public Key Model下安全地压缩公钥和签名值，并且验证签名过程等同于普通的Schnorr签名验证。Bitcoin Cash网络已经激活了Schnorr签名机制，为后续部署MuSig方案铺平了道路。Bitcoin网络中则围绕Schnorr签名和MuSig签名机制准备了一揽子升级计划 [26, 27, 28, 29]。MuSig签名机制的优点在于可以基于secp256k1进行构建，而目前的Cosmos SDK已经支持secp256k1曲线。Boneh等人在2018年构建了在Plain Public Key Model安全的BLS聚合签名算法 [15]，同样可以用于多签交易的改进。为了支持BLS聚合签名算法，要求重构钱包体系 [30]，这是对现有系统的深层改动。从secp256k1的私钥切换到某个双线性对友好的椭圆曲线上的私钥（例如Zcash团队构建的BLS128-381曲线 [31]），需要同时修改分层确定性钱包的实现。因此，基于MuSig机制改进多签交易目前看来是更兼容当前系统的方式。

Blockstream研究人员设计的MuSig多签机制 [14] 与基于双线性对的BLS签名机制 [15] 理想的情况是可以对一个块中所有交易的签名进行压缩，压缩之后区块中仅有一个签名值待验证。这需要聚合签名算法能够把不同私钥对不同消息的签名值聚合起来。这种情况下，矿

工在打包区块时可以聚合所有待包含交易的签名。MuSig无法对不同的消息进行聚合，虽然论文[14]中提到可以通过修正的交互聚合签名（Fixed Interactive Aggregate Signature）机制实现对不同消息的签名聚合，但是尚没有严谨的安全性证明。论文[15]在更强的安全性假设基础上构建的聚合多签机制（Aggregate Multisignature Scheme，AMSP）可以用来聚合不同交易的签名以进一步节省链上的存储空间。在更强的安全性假设之外，如何在区块链场景下安全地部署这一机制也需要进一步探索。

区块链

Tendermint共识协议不分叉和逐块最终化的特性简化了区块链结构的设计，只需要用哈希指针一次串联起各个区块，无需像比特币一样考虑区块的回滚也不需要像Ethereum中考虑对叔块的连接，每个区块也由区块头和区块体两部分构成。区块头中包含区块高度，时间戳，区块中交易个数，累积的交易个数，指向上一个区块的哈希指针，本区块的提案者，包含的证据列表的Merkle树根，包含的交易列表的Merkle树根，针对上一个区块的投票列表的Merkle树根，上一个区块的验证者集合的Merkle树根，上一个区块所产生的新的验证者集合的Merkle树根，上一个区块执行完成后上层应用内部状态的Merkle树根，上一个区块中交易执行后的结果列表的Merkle树根以及共识参数的Merkle树根。区块体中则包含了具体的交易信息，证据信息以及对上一个区块的投票信息。



值得注意的是，针对一个区块的投票信息存储在下一个区块中。Tendermint共识协议中的投票本质上是验证者用自己的私钥对一个区块的签名值，当前采用的是基于Ed25519曲线的EdDSA签名算法[32]。随着验证者集合的增多，这些投票/签名信息也会占据链上可观的存储空间而且验证签名也需要耗费可观的计算资源。针对计算资源消耗问题，可以通过EdDSA签名机制本身支持的批量验证进行改进。要同时减少签名值占用的存储空间和计算资源消耗的问题则可以借助聚合签名机制。基于MuSig的聚合签名机制，在签名时各个签名参与方之间需要多轮交互，考虑到验证者可能遍布全球各地，我们倾向于避免在共识协议必需

的交互之外引入更多的交互需求，也因此基于MuSig的聚合签名机制不宜作为共识过程中的聚合签名机制。与交易的聚合签名不同的是，投票的签名聚合功能上更为独立，不受钱包的分层密钥等其他业务逻辑的牵连，也因此可以考虑全新的签名机制。这种情况下，更适合采用[15]中提出的聚合BLS签名机制来改进Tendermint Core项目的投票机制。

为了能够更好地支持跨链和轻客户端，每个区块中还包含了上层应用内部状态的Merkle树根。Merkle树属于可验证数据结构（Authenticated Data Structure，ADS），为了支持轻客户端的快速验证，需要能够支持存在性证明与不存在证明。而状态需要逐块更新的特性也要求ADS数据结构支持增量增加和修改，CoinEx Chain目前采用的是Tendermint Core项目自带的基于数据结构IAVL+的ADS。

基于IAVL+的ADS与Ethereum的Merkle Patricia Tree（MPT）相比并没有显著优势。基于Ethereum的经验可知，上层业务的状态读写会被MPT数据结构放大多倍（从根节点出发逐步遍历到叶子节点的过程需要多次访问底层的键值对数据库），从而成为潜在的性能瓶颈。ADS数据结构引入的读写放大的问题，可以通过更好的工程手段进行缓解，也可以通过设计全新的ADS结构来应对。可能的解决方案是开发一种全新的ADS，它独特的树结构允许中间节点被保存在KV数据库之外，同时可以在程序异常重启后方便地从KV数据库恢复出中间节点，而不必担心数据一致性的问题。通过把常用的中间节点缓存在内存中，不常见的中间节点保持在磁盘上，确保大多数键值对访问只需要访问内存中的中间节点，从而大幅度减轻底层KV数据库的压力。

私钥安全

数字货币领域的资产的所有权是通过私钥进行签名来界定的，创建一笔交易通常需要访问私钥，而私钥的保护一直是个困难问题。数字钱包中私钥通常是通过keystore文件进行加密存放，加密该私钥的密钥通常由用户口令等通过密钥派生算法生成。虽然keystore文件仅存储私钥的密文，但是创建交易的时候仍然需要将其解密，也即完整的私钥信息总会暴露在系统中，同时也就有了泄露的风险。也可以采用冷钱包的策略，通过在需要时引入人工协助的方式将系统隔离开，从而降低私钥泄露的风险。通过离线存储，冷钱包降低了私钥被恶意软件窃取的可能性，是一种通过管理手段提升安全性的方法。

安全性要求更强的场景通常使用硬件安全模块（Hardware Security Module）进行密钥的防护，通过HSM模块生成私钥、计算签名可以保证明文私钥信息不会存在于HSM模块之外，从而保证私钥的安全性。冷钱包方案和HSM方案能够提升私钥的安全性，但也带来了使用的不便。而当涉及到多签交易或者需要将私钥的访问权限分散到多个成员时，要求每一方都使用冷钱包或者HSM提高了使用门槛，也进一步增加了使用的复杂性。另外Ledger的研究人员将在Black Hat 2019上展示的对HSM的攻击[33,34]，也说明HSM自身也存在安全隐患。

多签交易通过要求多个签名来保证授权一笔交易，就需要多个私钥共同配合的方式，攻击者同时需要多个密钥信息才能够转移资产，作恶的难度加大。另外通过调整m和n的参数，多重签名机制也提供了一定程度的安全冗余，只要被窃取或者丢失的的密钥个数小于n-m依然可以构造交易。然而多签交易的交易费会更高，变更安全策略也不够灵活，另外所有的签名值和公钥会公开在链上，导致签名策略的泄露。

Shamir秘密分享方案[11]可以解决多签交易的问题。在SSS方案中，资产的管理权仅由一个私钥控制。资产的转移与标准交易一样只需要一个签名值。然而这个私钥是由多个参与方控制的，这是通过把私钥分割成多份并且在多个参与方之间分发这些密钥分片来做到的。获得足够多的密钥分片后，可以重构出原始的私钥。与多签交易类似，也可以选择m-of-n的SSS方案。m-of-n的SSS方案中需要至少m个密钥分片才能重构出原始密钥，而通过m-1或者更少的密钥分片无法获得原始密钥的任何信息。m-of-n的SSS方案与m-of-n的多签方案一样增强了密钥安全性并能容忍密钥（分片）遗失。SSS方案中重构原始密钥的操作需要由某一方来完成。被选中的一方在完成原始密钥重构之后会获得原始密钥的所有信息。如果采用SSS方案，则必须相信被选中的一方在使用密钥之后会擦除并且遗忘该密钥的信息。很不幸，在尝试将密钥控制分散化的过程中，再次引入了可信第三方和单点失败的问题。

来自安全多方计算领域的研究进展带来的多方阈值ECDSA签名机制（{m,n} threshold ECDSA）[12, 13]同时具备Shamir秘密分享方案与多签交易两种方法的优势：

1）支持与Shamir秘密分享方案类似的密钥分割，但是在签名时无需重构原始密钥信息，这也就避免了Shamir秘密分享方案中单点失败的问题；
2）与多签交易相比，最终体现在链上的相关信息与传统P2PKH交易相同，只需要一个签名值，没有增加交易的体积也无需暴露访问控制策略，由于此时的交易与普通的交易无法根据链上信息来区分，所以也提供了更好的隐私属性。

CoinEx Chain会在钱包中支持ECDSA的阈值多方签名机制，作为私钥保护的增强方案。

DEX

CoinEx DEX公链是基于Tendermint共识协议打造的专门用于去中心化交易的公链，在DEX公链上，用户可以收发CET代币，发行新的代币并进行增发，燃烧，锁定，解锁等操作，以及创建交易对，交易下单, 查询交易历史，竞选成为验证者节点等。

通过将资产的控制权返还用户，用户掌管自己的私钥，可以避免中心化交易所带来的单点失败风险，而CoinEx Chain提供的多方阈值ECDSA签名机制能够进一步增强用户侧的私钥保护。借助Tendermint共识协议以及精简的链上功能实现秒级出块速度与瞬时交易确认。通过资产上链、链上交易、链上撮合的策略，实现公平透明的交易体验。 区别于传统的智能合约发币，DEX公链内置发币模块，使发币更加高效和安全。用户无需许可即可发布Token（包括但不限于稳定币），同时可创建该Token相关交易对，省去中心化交易所冗长的审核流程和高昂上币费用。DEX公链上每一步操作都是标准化的，每一步操作消耗的资源都是可预见的，因此DEX公链可做到高达每秒数千笔的交易处理速度。

CET通过主网映射的方式发行，CET除了可以作为链上交易的手续费之外，还可以作为抵押代币，CET持有者可参与CoinEx Chain中的staking经济。另外CET持有者也可进行提案的发起和投票，参与社区治理。

CET主网映射

CoinEx基金会和CoinEx商业生态将协作完成CET的主网映射，CET持有者需要把ERC20 CET充值到CoinEx交易所，主网上相应的CET会分发给CoinEx交易所，随后用户可在CoinEx交易所提取主网代币CET，具体流程如下：

1）在主网上线前，交易所只允许充值ERC20 CET ，不允许提现
2）所有非锁仓的币都是用户存在CoinEx交易所的币
3）主网在启动时，会将CoinEx交易所中的ERC20 CET以及锁仓的ERC20 CET做主网映射
4）开通主网CET的提现和充值
5）用户创建CoinEx Chain主网帐户地址后，可从CoinEx交易所提现主网CET
6）用户也可以提现到第三方钱包或者独立钱包，使用主网CET参与Staking

CET分发

CET主网映射后的分布如下：

持有者	账号类型	金额	解锁时间（UTC）
普通用户	普通账号	28.88亿	已经流通
CoinEx基金会	普通账号	8.85亿	按需解锁
CoinEx基金会	普通账号	3.15亿	预留的出块激励
CoinEx团队	锁仓账号	3.6亿	2020/1/1
CoinEx团队	锁仓账号	3.6亿	2021/1/1
CoinEx团队	锁仓账号	3.6亿	2022/1/1
CoinEx团队	锁仓账号	3.6亿	2023/1/1
CoinEx团队	锁仓账号	3.6亿	2024/1/1

CET激励

如前所述，CoinEx会信守不增发CET的承诺，因此不会采用通胀的方式铸造新币。然而区块激励对于提高社区参与度至关重要，因此在主网上线后，CoinEx基金会将分配约3.15亿个CET，用于激励初期验证节点及staking参与者。预留CET激励发放的总体时长与出块间隔时间有关，激励计划按出块速度为3s进行估算，每个区块的具体奖励金额参见下面的表格：

	开始高度	结束高度	CET金额	每块激励
第1年	0	10512000	105,120,000	10
第2年	10512000	21024000	84,096,000	8
第3年	21024000	31536000	63,072,000	6
第4年	31536000	42048000	42,048,000	4
第5年	42048000	52560000	21,024,000	2

区块奖励之外，每个区块的收益还包括区块中交易的手续费。交易手续费包含两个部分：通常意义上的Gas费用以及功能费。Gas费用是为了防止对系统资源的恶意滥用，功能费部分主要用来提升链上生态的质量，防止恶意使用相关功能，确保用户体验。主网启动时会设置特殊操作的功能费，而后期可以根据主链演进情况，通过社区提案的方式对各个特殊

操作的功能费进行调整。特殊操作包括：新Token的发行，新交易对上线，锁定转账，新账户激活以及交易撮合。

Token发行与交易

在链上发行新Token是无需审查的，任何人都可以发行Token，也可以为发行的Token创建新的交易对，与发行新Token一样，创建新交易对也无需审查。为了保证新Token的流通性，为新发行的Token创建的第一个交易对必须是该新Token与CET之间的交易对。为了避免对系统资源的滥用，保证链上生态的质量，发行Token和创建新交易对会收取一定金额的CET作为功能费。Token的Symbol符号由2-8位字符数字组成，不可以数字开头。Token的精度为8位（十进制），Token发行最大数量为 900亿。Token发行者即为Token所有者（Owner），所有权可转移给他人。

发行Token时的可选项有燃烧、增发、冻结地址、冻结币种，并且这些选项只能在发行Token时指定，发行之后不可更改。如果Token发行时没有开启冻结地址和冻结币种的选项，则该Token的流转和持有都是自由且不受限制的。

开启冻结地址选项后，Token Owner可按需冻结部分地址，被冻结地址中的该Token无法转帐，也不能进行Exchange交易，但不影响地址中的其它Token。开启冻结币种选项后，Token Owner可按需对该Token进行冻结，冻结期间该Token的转账和Exchange交易被全局禁止。Token全局冻结期间Token Owner可以创建地址白名单，白名单中的地址可发起转账交易但不能进行Exchange交易。Token全局冻结期间，Token Owner对Token的操作不受影响。

	冻结币种		冻结地址
	普通地址	白名单地址 & Token Owner	被冻结的地址
创建order	✗	✗	✗
撮合order	✗	✗	✗
发起转账	✗	✓	✗
收款	✓	✓	✓

链上治理

CoinEx Chain的验证节点初始数量为42个，未达到验证节点数量上限时，任何人都可以通过发出CreateValidator交易来创建验证者。网络验证节点数量达到上限后，验证节点按质押的CET数量进行排序，选取质押量最高的42个验证者。

社区通过先提案、再投票的方式来达成社区治理，验证者可以替委托人投票，但是委托人也有权对验证者的投票进行覆盖重投.

投票时有四种选项：同意，弃权，反对，强烈反对:

- 如果有大于1/3的人投强烈反对票，则提案失败；
- 如果参与投票的质押代币没有达到所有质押代币的40% ，则提案失败；
- 如果非弃权票中有超过1/2同意票，则提案通过

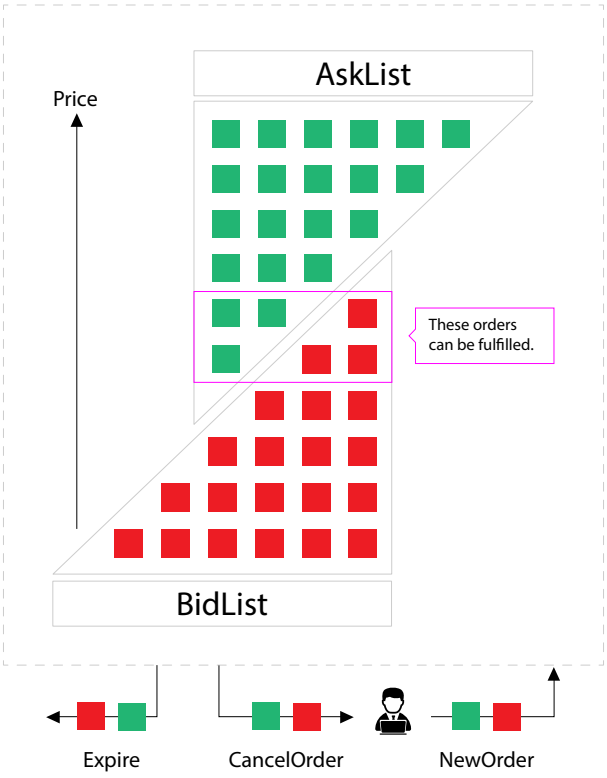
投票统计相关的比例值，都由参数进行配置，后续会具备通过提案调整参数的能力。

发起提案后，社区需要向相关提案质押10000个CET代币以防止提案的滥用。提案通过后会退回给原充值帐户地址。提案因以下情况不通过时，相关的质押金额会被没收不再退回，没收的质押金额由系统保留将来用于社区激励。

- 充值未达到10000个CET，说明社区对此提案不感兴趣或不支持
- 如果参与投票的质押代币没有达到所有质押代币的40% ，则提案失败；
- 有大于1/3的人投强烈反对票

交易撮合

同主流的中心化交易所一样，我们采用基于订单簿的撮合方式，如下图所示：



订单簿包含卖单列表（AskList）和买单列表（BidList），卖单被标记为绿色，买单被标记为红色。卖单总是希望拉高价格，买单总是希望压低价格。目前只支持限价单，不支持市价单。如果卖一和买一的价格没有交叉的话，市场是没有成交的。

AskList和BidList内部的组织方式是，先按照价格排序，价格更好的单排在最前面，有资格优先成交；而价格相同的订单，更早被打包上链的，即年龄更大的，有资格优先成交。在图中，价格越低的卖单在AskList中排列得最靠队列头（靠下），而价格越高的买单在BidList中排列得最靠队列头（靠上）。

当卖一和买一的价格有交叉的时候，比卖一价格高的买单，以及比买一价格低的卖单，会先按价格后按年龄来排序，逐一成交。图中紫色方块所圈出的卖单和买单，都有资格参与排序，最终有可能成交。 最终它们是否能全部成交，还要看它们当中卖的总量和买的总量。

用户可以通过NewOrder交易提交新的卖单或买单到订单簿中，也可以随时撤销订单簿中由自己发出的订单。Good Till Expire（GTE）订单和 Immediate Or Cancel （IOC）订单在过期后都会自动地从订单簿中删除。前者会在预设年龄达到之后的UTC时间午夜零点过期，预设年龄可以通过支付更多的功能费来延长。IOC订单会在进入订单簿之后的下一个区块过期（即只有一次被撮合的机会）。

链下撮合，订单总是逐一被服务器接受的，可以对它们按先来后到排出全序。在链上撮合最重要的不同点是，订单是批量被打包上链的，对同一个区块内部的订单，无法对它们进行先来后到的排序。为了保证对同一个区块内部的订单一视同仁，我们采用“集合竞价”的方式：每轮撮合，针对可以成交的所有买单和卖单，计算出一个单一的执行价格。决定执行价格的原则是：

1. 使成交量最大化。
2. 使剩余量最小化。如果有超过一个的价格都能实现相同的交易量，则选择剩余量最小化的价格。剩余量是指可接受执行价格的订单中，残余下来未成交的数额。
3. 市场压力。如果多个价格都能满足上述第1步和第2步的要求，那么先确认目前市场对潜在价格的压力是哪个方向的。如果剩余量是正数，则买方压力大，选择更高的价格；如果剩余量是负数，则卖方压力大，选择更低的价格。
4. 当正的剩余量和负的剩余量同时存在时，以上一次成交时的执行价格为参考价格，选择距离参考价格最近的价格。

无论是中心化交易所还是去中心化交易所，都需要应对抢先交易的问题。抢先交易是指借助技术优势或者市场优势而预先获知关于交易的信息，提前预测成交价的变动并据此执行对自己有利的交易，这往往会使得市场的其他参与方蒙受损失。例如在中心化交易所场景

下，交易所能够看到更为全局的交易信息，从而在撮合交易之前，根据当前市场情况为自己炮制最优交易策略并优先成交自身交易以牟利。

CoinEx的DEX公链在设计上，天然地具有防止抢先交易的特性。首先，基于Tendermint的秒级出块速度，使得进行抢先交易的时间窗口很小；其次，在P2P网络中，很难完整了解到究竟哪些订单会被包括到下一个区块中；再次，集合竞价的价格生成机制，让抢先的交易和同一个区块内的其它交易相比，不具有明显优势。因此，普通用户希望通过抢先交易牟利是非常困难的。

作为Validator，有权决定下一个区块当中新增哪些订单，它可以通过“审查攻击”的方式，在区块中包含自己专门设计的订单，而不包含那些同自己的订单有冲突的订单，以此来侵占其它交易者的利益。但是，DEX上的所有数据和执行逻辑都是公开的，如果Validator频繁地对他人的交易进行审查攻击，同时插入并未经过全网P2P广播的交易，那么它的行为会很容易地被观察到。这会损害它的信用，最终导致委托人撤销对它的支持。

一项可能的改善是，通过“Commit-Reveal”机制，允许用户选择性地让订单的内容延迟揭示（例如延后2~3个块），这样当验证者打包下一个区块的时候无法看到当前交易系统所有的状态，也使得想要构造能够获利的抢先交易更为困难。CoinEx团队会持续深入研究抢先交易问题，并在后续提供更为完善的解决方案。

自动化做市

目前交易所广泛采用订单簿的撮合方式，要达成交易，需要同时有买卖的需求，并且订单薄上的AskList和BidList必须有交集，也就是说最少买一的价格要大于等于卖一。流动性好的币种比如BTC，因为买一或者卖一的订单很大，因此如果想买卖一定量的话基本上在交易所能够满足需求，但是有些关注度小的币种，其流动性不足以支撑大额的买卖单。基于这种情况，很多项目会选择做市商增加其流动性，做市商通过做市制度来维持市场的流动性，他们通过买卖报价的适当差价来补偿所提供服务的成本费用，赚取利润。

做市的成本高是Token发行方面临的一大问题，一方面是交易所要收取手续费，一方面是做市商要获取利润，这些都是增加流动性所带来的成本。另外，做市商做市还是要依靠真实的交易量，虽然做市商都是程序运行，当出现单边大量买卖单的时候，传统的做市商可以提供的做市能力非常有限。CoinEx Chain会在DEX公链上使用自动化算法做市的方式来满足代币的流动性，主要基于两种自动化做市协议实现：Bancor [35]和UniSwap [36,37]。CoinEx Chain会对Bancor协议和UniSwap进行扩展，使其更加适应去中心化交易所，更好地提供足够的流动性，并且保证其价格的合理性。做市商在提供流动性时不再指定交易价格，而是只提供资金。

Bancor协议能够实现具有去中心化流通性的Token交易网络，协议不依赖于双边需求匹配，运用连接器（Connector）实现异步价格机制。Token的价格公式为：

price=connectorBalance/(smartTokenSupply * CW)，其中连接器权重（CW）影响smartToken价格对smartToken供给的敏感度。用户可以随时按照自动计算的价格从连接器买到Token，也可以随时按照自动计算的价格把Token卖给连接器。

UniSwap也是一个去中心化Token交易所协议，Uniswap完全摆脱限价订单的概念，UniSwap会根据乘积恒定做市商模型做市，将每个人的流动性汇聚在一起。UniSwap会随着成交的买单增加，渐进式增加代币的价格，会随着成交的卖单增加，渐进式降低代币的价格。通过算法可以实现不需要挂单，不需要市场深度的去中心化兑换系统。乘积恒定的核心思想是：x*y=k，其中x是Base Currency 数量，y是Quote Currency 数量，而k是两个数量的乘积。当保证k是一个固定不变的常量时，x值越大，y值就越小；x值越小，y值就越大。

多链与跨链

Smart Chain

为了保证DEX公链的交易处理速度，DEX公链仅支持去中心化交易所必备的功能。然而以太坊的经验已经体现出，智能合约是构建DeFi科技不可或缺的组件。因此为了构建CoinEx Chain完整生态，实现可编程现金的目标，基于特定应用公链的理念，CoinEx Chain团队将打造一条专门支持智能合约的公链。

Privacy Chain

隐私保护是区块链领域面临的巨大挑战。没有人希望自己的经济活动能够被追踪。比特币和以太坊的伪随机地址无法保护一个实体的活动不被追踪。已经有大量的链上隐私保护技术被提出并且其中一些已经在区块链项目中得到应用。当前主要的支持隐私保护的数字货币项目有Dash [38]，Monero [40]，Zcash [44] 以及 Grin/Beam [46,47]等。

Dash利用混币策略[39]实现交易的混淆以提升链上交易的隐私性；Monero利用Pedersen Commitment和范围证明[41,42]来隐藏交易金额，利用可链接环签名机制[43]来隐藏交易的发起方并同时保留检测双花的能力，利用Stealth Address来隐藏交易的接收方；ZCash利用zkSNARK [45]同时隐藏交易金额、发起方、接收方；Grin/Beam基于MimbleWimble协议[48]同样利用Pedersen Commitment和范围证明实现交易金额的隐藏，并利用两方交互Schnorr签名提升链上隐私属性。这些项目是基于UTXO模型的，而且随着密码学方案的改进，各个方案的成熟度、可用性也越来越高。

相比之下，在账户模型下进行交易的隐私保护的成熟方案相对较少。 以太坊在拜占庭升级中做了一次尝试，通过引入zkSNARK技术并借助智能合约，Baby ZoE将部分Zcash的隐私引入了以太坊中 [49]。

Zerochain项目[50,51]则尝试直接在账户模型下提供隐私保护功能：隐藏交易金额和账户余额。Zerochain是基于Substrate [54]区块链框架开发的原型验证，采用的密码学工具是具有加法同态属性的Lifted-ElGamal加密[55]以及基于Groth16 [52]方案的zkSNARK方案。具有加法同态属性的加密机制一方面是为了隐藏交易金额和账户余额，另一方面也是为了能够在密文状态下更新账户余额，而zkSNARK则以零知识的方式证明转账交易的合法性。Zerochain以PoC的形式验证了前述方案的可行性，但尚未探讨账户模型下匿名交易可能涉及的诸如重放保护以及front running等问题的应对措施。

来自斯坦福和Visa研究部门的研究人员提出的Zether方案[56]则能够在账户模型下同时隐藏交易金额以及交易发起方和接受方，并完整讨论了重放保护与front running问题的应对

措施。Zether同样采用了Lifted-ElGamal加密方案[55]实现对交易金额和账户余额的隐藏，并通过Bulletproof的改进版Σ-Bullets来进行零知识证明。Zether协议在论文[56]中实现为以太坊上的智能合约，但也可用来构建账户模型的匿名数字货币。另外通过扩展Zether协议，还可以将交易的发送方和接收方隐藏在由发送者选定的一组用户当中。Zether协议与区块链所基于的共识协议相互独立。这意味着结合Cosmos SDK的开发框架和Tendermint Core项目，可以构建基于Zether的匿名数字货币。CoinEx Chain团队会持续关注账户模型下隐私保护方案的进展，利用前沿技术构建基于Tendermint共识协议的具有隐私保护功能的账户模型公链。值得注意的是JPMorgan已经在这个方向做了尝试：在Ethereum的Quorum区块链中做了整合Zether协议的尝试 [57]。

跨链

世界上不会只有单一的一条链来满足人类的所有需求，多条定位不同、用途各异的链并存，不但是目前的常态，也会是未来的发展方向。而多条异质链相互之间通过兼容的跨链协议进行通讯，将打通信息和价值在不同链间流转的藩篱，极大地促进应用的发展。

Vitalik Buterin在论文[58]中总结到，实现跨链有三种基本的机制：哈希锁、公证人和中继。哈希锁原理清晰，实现简单，已经在闪电网络[59]等支付通道中得到广泛应用。公证人机制做不到完全“去信任”，是相对中心化的方案，但在实现同已有链（尤其是缺乏Finality的PoW链，例如Bitcoin和Ethereum）的互连时，是最为可行的方案，也得到了一定的应用。

中继机制是最符合去中心化要求的方案，也是目前Cosmos和Polkadot两个备受关注的跨链项目所采用的技术。Cosmos的链间通信协议（Inter-Blockchain Communication，IBC）在资产跨链之外也支持数据跨链等更为丰富的功能，与CoinEx Chain所基于的Tendermint Core以及Cosmos SDK具有更好的兼容性，CoinEx Chain在跨链通信方面将选择基于IBC的方案。

基于哈希锁实现的原子交换作为一种成熟且容易实现的资产跨链机制，也将在CoinEx Chain上得到应用，例如：一、利用它在CoinEx Chain上发行锚定其它数字货币的代币，例如BTC、ETH，用户同代币发行方在原生链上和CoinEx Chain上交换数字货币以及与之锚定的代币；二、代币发行者利用它在多条链上发行同质的资产，就像USDT同时在Bitcoin、Ethereum、EOS和Tron上发行，发币方可以同时在以太坊、CoinEx Chain甚至其他DEX公链上发行同质的资产，用户通过与发币方进行原子交换，能够将自己的代币在不同链之间进行迁移。

结论

CoinEx Chain致力于打造下一代区块链金融基础设施，是为支持可编程现金而打造的一系列公链项目，目前的规划中包含三条特定应用方向的公链：

1）支持去中心化交易功能的DEX公链；
2）支持智能合约功能的Smart公链；
3）支持链上隐私保护功能的Privacy公链。

三条公链之间通过IBC链间通信协议进行互联互通，各司其职又相互配合提供完备的功能。

DEX公链通过资产上链、链上交易和链上撮合等功能，解决目前中心化交易所被广泛诟病的安全性差、不透明等问题。通过将资产控制权归还用户、基于订单簿的公平的链上撮合算法、无需许可的上币以及创建交易对的功能，DEX公链旨在构建透明、安全、无许可的自由金融平台。底层的Tendermint共识提供的高TPS以及秒级交易确认的特性，能够在最大程度上还原中心化交易所的体验。

为了最大限度地提高交易处理性能，DEX公链选择仅实现必要的功能。然而，功能更为丰富的金融应用依赖功能完备的智能合约，而链上隐私保护也日益成为安全焦点。因此，在DEX公链之外，CoinEx团队还将打造一条支持智能合约功能的Smart公链以及支持链上隐私保护功能的Privacy公链：Smart公链为CoinEx Chain生态提供智能合约支持，为构建复杂的金融应用提供平台；Privacy公链基于最新的密码学进展，例如Zether协议在账户模型下提供交易金额、发起方以及接收方的信息隐藏。

DEX公链、Smart公链以及Privacy公链不是相互隔离的孤岛，通过基于中继的跨链通信解决方案实现互联互通，功能互补。需要参与复杂金融合约的CET代币可以通过DEX公链转移到Smart公链，并在结束之后再转回DEX公链。而需要参与代币混淆的CET代币也可以通过Privacy公链的隐私交易进行，并可以最终再返回DEX公链。这样三条公链各司其职，在保证各自的交易处理速度和功能属性之外，也可以联合提供更为丰富更为安全的功能。在将来，CoinEx团队也会根据社区需求继续打造特定应用的公链以进一步丰富CET的生态体系。

CoinEx团队同时会对现有的技术基础设施进行优化，这包括尝试利用增量VDF机制改进基于PoS机制的公链的安全性，利用MuSig方案来改进多签交易，减少链上空间占用并增强多签及交易的隐私属性。利用聚合BLS签名方案改进Tendermint共识协议的投票过程，减少投票信息所占用的链上存储空间。另外，CoinEx团队会通过阈值多方ECDSA签名机制为用户的私钥保护提供更理想的解决方案。

参考文献

1. Bitshares Blockchain. Open-source business development and financial management platform. https://bitshares.org/
2. EtherDelta. https://etherdelta.com/
3. 0x Protocol. Powering Decentralized Exchange. https://0x.org/
4. Joseph Poon, and OmiseGO Team. OmiseGO：Decentralized Exchange and Payments Platform. 201706. https://cdn.omise.co/omg/whitepaper.pdf
5. Daniel Wang, Jay Zhou, Alex Wang, and Matthew Finestone. Loopring：A Decentralized Token Exchange Protocol. 201809. https://loopring.org/resources/en_whitepaper.pdf
6. Kyber：An On-Chain Liquidity Protocol v0.1. 201904. https://files.kyber.network/Kyber_Protocol_22_April_v0.1.pdf
7. Ethan Buchman. Tendermint：Byzantine Fault Tolerance in the Age of Blockchains. 201606. https://allquantor.at/blockchainbib/pdf/buchman2016tendermint.pdf
8. Ethan Buchman, Jae Kwon, and Zarko Milosevic. "The latest gossip on BFT consensus." arXiv preprint arXiv:1807.04938 (2018). https://arxiv.org/pdf/1807.04938.pdf
9. Cosmos SDK. Blockchain Application Framework. https://cosmos.network/docs/
10. Jae Kwon, and Ethan Buchman. Cosmos：A Network of Distributed Ledgers. https://cosmos.network/cosmos-whitepaper.pdf
11. Adi Shamir. How to share a secret. Communications of the ACM 22, no. 11 (1979)：612-613.
12. Yehuda Lindell. Fast secure two-party ECDSA signing. In Annual International Cryptology Conference, pp. 613-644. Springer, Cham, 2017. https://eprint.iacr.org/2017/552.pdf
13. Rosario Gennaro, and Steven Goldfeder. Fast multiparty threshold ecdsa with fast trustless setup. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, pp. 1179-1194. ACM, 2018. https://eprint.iacr.org/2019/114.pdf
14. Gregory Maxwell, Andrew Poelstra, Yannick Seurin, and Pieter Wuille. Simple schnorr multi-signatures with applications to bitcoin. Designs, Codes and Cryptography (2018)：1-26. https://eprint.iacr.org/2018/068.pdf
15. Dan Boneh, Manu Drijvers, and Gregory Neven. Compact multi-signatures for smaller blockchains. In International Conference on the Theory and Application of Cryptology and Information Security, pp. 435-464. Springer, Cham, 2018. https://eprint.iacr.org/2018/483.pdf
16. IAVL+ Tree：Merkleized IAVL+ Tree implementation in Go. https://github.com/tendermint/iavl

17. Gavin Wood. Ethereum：A Secure Decentralised Generalised Transaction Ledger Byzantium. Version d6ff64f - 2019-06-13. https://ethereum.github.io/yellowpaper/paper.pdf.
18. Seth Gilbert, and Nancy Lynch. Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. Acm Sigact News 33, no. 2 (2002)：51-59. https://users.ece.cmu.edu/~adrian/731-sp04/readings/GL-cap.pdf
19. Vitalik Buterin. On Stake. 201407. https://blog.ethereum.org/2014/07/05/stake/
20. Vitalik Buterin. Long-Range Attacks：The Serious Problem With Adaptive Proof of Work. 201405. https://blog.ethereum.org/2014/05/15/long-range-attacks-the-serious-problem-with-adaptive-proof-of-work/
21. Vitalik Buterin. Proof of Stake：How I Learned to Love Weak Subjectivity. 201411. https://blog.ethereum.org/2014/11/25/proof-stake-learned-love-weak-subjectivity/
22. Dan Boneh, Joseph Bonneau, Benedikt Bünz, and Ben Fisch. Verifiable delay functions. In Annual International Cryptology Conference, pp. 757-788. Springer, Cham, 2018. https://eprint.iacr.org/2018/601.pdf
23. Benjamin Wesolowski. Efficient verifiable delay functions. In Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 379-407. Springer, Cham, 2019. https://eprint.iacr.org/2018/623.pdf
24. Krzysztof Pietrzak. Simple verifiable delay functions. In 10th Innovations in Theoretical Computer Science Conference (ITCS 2019). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018. https://eprint.iacr.org/2018/627.pdf
25. Naomi Ephraim, Cody Freitag, Ilan Komargodski, and Rafael Pass. Continuous Verifiable Delay Functions. 201706. https://eprint.iacr.org/2019/619.pdf
26. Johnson Lau. BIP114：Merelized Abstract Syntax Tree. 201604. https://github.com/bitcoin/bips/blob/master/bip-0114.mediawiki
27. Pieter Wuille. BIP draft：Schnorr Signatures for secp256k1. https://github.com/sipa/bips/blob/bip-schnorr/bip-schnorr.mediawiki
28. Gregory Maxwell. [bitcoin-dev] Taproot：Privacy preserving switchable scripting. 201801. https://lists.linuxfoundation.org/pipermail/bitcoin-dev/2018-January/015614.html
29. Gregory Maxwell. [bitcoin-dev] Graftroot：Private and efficient surrogate scripts under the taproot assumption. 201801. https://lists.linuxfoundation.org/pipermail/bitcoin-dev/2018-February/015700.html
30. Pieter Wuille. BIP32：Hierarchical Deterministic Wallets. 201202. https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki
31. Sean Bowe. BLS12-381：New zk-SNARK Elliptic Curve Construction. 201703. https://electriccoin.co/blog/new-snark-curve/
32. Simon Josefsson, and Ilari Liusvaara. RFC8032. Edwards-Curve Digital Signature Algorithm (EdDSA). 201701. https://tools.ietf.org/html/rfc8032

33. Gabriel Campana, and Jean-Baptiste Bédrune. Everybody be Cool, This is a Robbery! https://www.blackhat.com/us-19/briefings/schedule/?hootPostID=db681a52c6a321681e1f9281b5124457#everybody-be-cool-this-is-a-robbery-16233
34. Graham Steel. How Ledger Hacked an HSM. 201906. https://cryptosense.com/blog/how-ledger-hacked-an-hsm/
35. Eyal Hertzog, Guy Benartzi, and Galia Benartzi. Bancor Protocol：Continuous Liquidity for Cryptographic Tokens through their Smart Contracts. 201803. https://storage.googleapis.com/website-bancor/2018/04/01ba8253-bancor_protocol_whitepaper_en.pdf
36. Uniswap Whitepaper. https://hackmd.io/C-DvwDSfSxuh-Gd4WKE_ig
37. Yi Zhang, Xiaohong Chen, and Daejun Park. Formal Specification of Constant Product (x × y = k) Market Maker Model and Implementation. 201810. https://github.com/runtimeverification/verified-smart-contracts/blob/uniswap/uniswap/x-y-k.pdf
38. Dash - Dash is Digital Cash You Can Spend Anywhere. https://www.dash.org/
39. Gregory Maxwell. CoinJoin：Bitcoin privacy for the real world. 201308. https://bitcointalk.org/index.php?topic=279249.0
40. Monero - secure, private, untraceable. https://www.getmonero.org/
41. Gregory Maxwell, and Andrew Poelstra. Borromean ring signatures. 201506. https://pdfs.semanticscholar.org/4160/470c7f6cf05ffc81a98e8fd67fb0c84836ea.pdf
42. Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. Bulletproofs：Short proofs for confidential transactions and more. In 2018 IEEE Symposium on Security and Privacy (SP), pp. 315-334. IEEE, 2018. https://eprint.iacr.org/2017/1066.pdf
43. Shen Noether. Ring Confidential Transactions for Monero. IACR Cryptology ePrint Archive, 2015, p.1098. https://eprint.iacr.org/2015/1098.pdf
44. Daira Hopwood, Sean Bowe, Taylor Hornby, and Nathan Wilcox. Zcash Protocol Specification. Version 2019.0.2 [Overwinter+Sapling]. 201906. https://raw.githubusercontent.com/zcash/zips/master/protocol/protocol.pdf
45. Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza. Succinct non-interactive zero knowledge for a von Neumann architecture. In 23rd USENIX Security Symposium (USENIX Security 14), pp. 781-796. 2014. https://eprint.iacr.org/2013/879.pdf
46. Grin. https://grin-tech.org/
47. BEAM：Mimblewimble-based Privacy Coin. https://www.beam.mw/
48. Tom Elvis Jedusor. Mimblewimble. (2016). https://github.com/mimblewimble/docs/wiki/MimbleWimble-Origin
49. Christian Reitwiessner. An Update on Integrating Zcash on Ethereum (ZoE). 201701. https://blog.ethereum.org/2017/01/19/update-integrating-zcash-ethereum/

50. Osuke. Announcing Zerochain：Applying zk-SNARKs to Substrate. 201903. https://medium.com/layerx/announcing-zerochain-5b08e158355d
51. Zerochain：A privacy-protecting blockchain on Substrate. https://github.com/LayerXcom/zero-chain
52. Jens Groth. On the size of pairing-based non-interactive arguments. In Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 305-326. Springer, Berlin, Heidelberg, 2016. https://eprint.iacr.org/2016/260.pdf
53. Gavin Wood. Polkadot：Vision for a Heterogeneous Multi-Chain Framework. Draft 1.
54. Substrate：The foundation for blockchain innovators. https://www.parity.io/substrate/
55. ElGamal, Taher. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE transactions on information theory 31, no. 4 (1985)：469-472. https://link.springer.com/content/pdf/10.1007/3-540-39568-7_2.pdf
56. Benedikt Bünz, Shashank Agrawal, Mahdi Zamani, and Dan Boneh. Zether：Towards Privacy in a Smart Contract World. 2019. https://eprint.iacr.org/2019/191.pdf
57. Benjamin E. Diamond. Anonymous Zether：Technical Report. 201905. https://github.com/jpmorganchase/anonymous-zether/blob/master/docs/AnonZether.pdf
58. Vitalik Buterin. Chain Interoperability. 201609. https://static1.squarespace.com/static/55f73743e4b051cfcc0b02cf/t/5886800ecd0f68de303349b1/1485209617040/Chain+Interoperability.pdf
59. Joseph Poon, and Thaddeus Dryja. The Bitcoin Lightning Network：Scalable Off-Chain Instant Payments. 201601. https://lightning.network/lightning-network-paper.pdf

去中心化公链生态系统，为金融自由而生。