레이어제로: 신뢰할 수 없는 옴니체인 상호운용성 프로토콜

라이언 자릭 브라이언 펠레그리노 케일럽 배니스터

2021년 5월 26일

Abstract

블록체인의 확산으로 개발자들은 애플리케이션 기능 및 처리량, 보안 및 비용에 대한 요구사항을 기반으로 스마트 컨택을 실행할 수 있는 다양한 플랫폼을 얻게 되었습니다. 그러나 이러한 자유로움으로 인해 심각한 분열이 발생합니다. 즉, 각 체인은 격리되므로 사용자는 유동성과 제한 옵션을 격리하여 장벽이 있는 생태계 간에 유동성과 상태를 이동시켜야 합니다.

이 백서에서는 다양한 크로스 체인 애플리케이션 세트를 구축할 수 있는 강력한 로우 레벨 통신 프리미티브를 제공하는 최초의 trustless omnichain 상호 운용성 프로토콜인 LayerZero에 대해 설명합니다. 개발자는 이 새로운 프리미티브를 사용하여 신뢰할 수 있는 관리자나 중간 트랜잭션에 의존하지 않고도 크로스 체인 DEX 또는 multi-chain yield aggregator 등의 seamless 인터체인 애플리케이션을 구현할 수 있습니다. 간단히 말해, Layer Zero는 모든 체인에 걸쳐 Trustlessly 직접 트랜잭션을 가능하게 하는 최초의 시스템입니다. 체인을 자유롭게 이동할 수 있도록 함으로써 사용자는 분리된 체인의 애플리케이션을 최대한 활용하면서 액체 상태의 단편화된 포켓을 통합할 수 있습니다. Layer Zero를 사용하면 완전히 연결된 미래 옴니체인 생태계의 기반이 되는 네트워크 패브릭을 제공합니다.

1. Introduction

블록체인 개념의 핵심에는 분산, 투명성, 불변의 3대 축이 있습니다. 블록체인을 제어하는 단일 엔티티는 없으며 블록체인에 대한 액션은 검증 가능하며 되돌릴 수 없습니다. 이러한 기둥은 기업이 다른 기업을 신뢰하지 않고 행동할 수 있는 기반을 만든다. 이러한 신뢰 보장은 예를 들어, 피아트 통화에 비해 암호화 화폐가 매력적인 이유 중 하나이다.

모든 사용자와 모든 애플리케이션이 하나의 unified 블록체인으로 공존한다면 이 백서는 여기서 마무리 짓게 될 것입니다. 하지만, 블록체인의 유틸리티로 인해 고유한 복잡성과 요구사항을 가진 다양한 애플리케이션이 확산되었습니다. 다양한 기능성에 대한 수요는 전문 체인의 성장을 방해했다. 각 체인은 자체 에코시스템 내에서 애플리케이션 성장을 촉진하고 있지만 이러한 에코시스템 간의 고립은 채택의 주요 제한으로 대두되고 있습니다. 사용자와 개발자는 시간, 리소스 및 유동성을 별도의 체인 간에 분할해야 합니다.
이른바 레이어 1 블록 체인의 수 (작성 시점에서는 109개까지 [1])의 극히 많은 수의 결과로서, 상기의 3개의 기둥을 복수의 체인에 걸쳐 동시에 Envelope 상호작용으로 확장할 필요가 있습니다. 인디맨드 인터랙션의 예로서 토큰의 전송이 있습니다. 이것은 이 섹션의 후반부에서 설명합니다.

블록체인의 용어로 작업단위는 Transaction이며, 불변하며 되돌릴 수 없습니다. 블록으로 정리된 트랜잭션은 블록 체인 시스템에서 보안의 기초를 형성합니다. 단, 트랜잭션은 항상 단일 체인 개념이었습니다. 아래에서 설명한 바와 같이 체인 간 상호 작용에는 통상적인 블록 체인 cryptosystem 이외의 서드파티 메커니즘이

필요했습니다. 이와는 대조적으로, 이 문서에서는 네이티브 교차 체인 트랜잭션이 가능한 최초의 메시징 프로토콜에 대해 설명합니다. : 레이어 제로

레이어 제로(Layer Zero)가 제공하는 강력한 통신을 설명하기 위해 토큰을 체인 간에 전송하는 예를 살펴봅니다. 현재, 2개의 체인의 토큰을 변환할때, 유저는 Centralized Exchange or Cross-chain decentralized Exchange (DEX)(크로스 체인 브리지라고도 함)를 활용합니다. 이것들은 모두 타협이 필요합니다.

예를 들어 Binance.com[3]와 같은 중앙 집중형 거래소의 경우, 사용자는 예금 및 자금 인출 내역을 추적하는 거래소를 신뢰해야 합니다. 이 신뢰관계는 블록체인 컨센서스의 근본적인 신뢰성과 배치되며 온체인 자동화 시스템의 보안성이 결여되어 있습니다.

AnySwap [2] 또는 TOR-Chain [23]과 같은 DEX를 사용하면 온 체인으로 전송을 수행함으로써 신뢰 문제가 완화되지만, 기존 DEX implementation 에서는 사용자 토큰을 특정 프로토콜 토큰으로 convert하는 것을 포함합니다 그 특정 프로토콜은 Transaction consensus를 얻기 위해 그들의 intermediate consensus layer를 통과합니다. 이 intermeiate consensus layer는 보통 안전한 방법으로 구현되지만 토큰 전송을 용이하게 하기 위해서는 사용자가 사이드 체인을 신뢰해야 합니다. 이 문서에서 알 수 있듯이 추가 오버헤드는 불필요합니다. 많은 사용자의 요구에도 불구하고, 효율적이면서도 직접적이면서도 애당초 블록체인을 만드는 핵심 이유인 신뢰성이라는 솔루션은 나타나지 않았습니다. 한 걸음 물러서서, Layer Zero의 직접적인 크로스 체인 트랜잭션은 개발자들에게 바로 그것을 구축하기 위한 도구를 제공합니다.

Layer Zero 및 위에서 설명한 이전 변경은 구현 스택의 두 가지 다른 수준에서 작동한다는 점에 유의해야 합니다. Layer Zero는 다양한 옴니체인 어플리케이션을 가능하게 하는 통신 프리미티브인 반면, Exchange는 Layer Zero 위에서 re-implementation함으로써 이익을 얻을 수 있는 어플리케이션의 한 예이다. 섹션 2에서는 블록 체인 테크놀로지의 개요를 설명하고, 그 교환의 예에 대해 설명합니다.

먼저, Layer Zero의 역량과 블록 체인 생태계에서 그 역할을 제대로 설명하기 위해, 우리는 valid delivery 라고 부르는 체인 간 트랜잭션을 가능하게 하기 위해 필요한 기본적인 커뮤니케이션 프라이머리티브의 공식화를 제시합니다(섹션 3).

다음으로 Layer Zero가 Trustless manner로 이 프리미티브를 제공함으로써 블록체인의 보안 약속을 유지하는 방법에 대해 설명합니다. 레이어 제로(Layer Zero)는 최초의 신뢰할 수 없는 옴니체인 상호운용성 레이어이며 레이어1 체인 및 레이어2 체인 간의 직접 메시징을 지원합니다(그림 1).

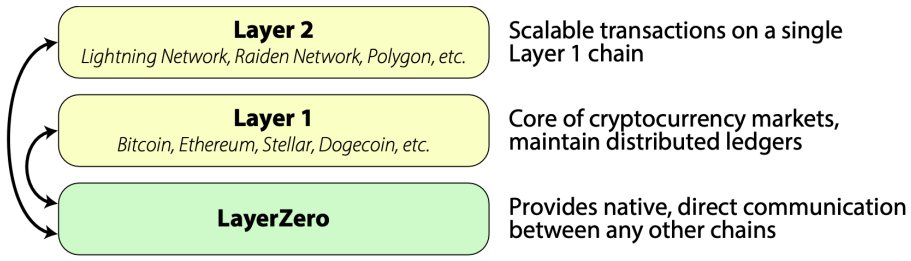

Figure 1: LayerZero enables cross-chain transactions.

체인 A와 체인 B 사이의 Cross-chain transaction은 다음으로 이루어져있습니다 - 1) A상의 transaction(tA), 2) A와 B 사이의 통신 프로토콜, 3) 메시지 m

Valid delivery 상태는 tA가 커밋되고 유효한 경우에만 m이 전달된다는 것입니다.

LayerZero를 뒷받침하는 핵심 아이디어는 두 개의 독립된 엔티티가 transaction(이 경우 tA)의 유효성을 입증하는 경우 체인 B는 tA가 유효하다는 것을 확신할 수 있다는 것입니다.

그림 2는 이것을 개략적으로 나타내고 있습니다. 담합하지 않는 두 개의 실체가 주어진 경우,

(1) 한 entity가 체인 A에서 tA를 포함하는 블록에 대한 블록 헤더를 생성할 수 있고,

(2) 다른 entity가 독립적으로 해당 블록의 tA에 대한 증명을 생성할 수 있으며(transaction proof),

(3) 헤더와 트랜잭션 증명에 실제로 동의하면,

Communication protocol은 체인 A에서 tA가 안정적으로 커밋되는 것을 보증하면서 체인 B의 클라이언트에 m을 전달할 수 있습니다.

섹션 4에서 설명된 LayerZero communication protocol은 수신자 체인의 트랜잭션이 어떠한 중간 체인을 포함하지 않고 송신자 체인의 유효하고 커밋된 트랜잭션과 쌍을 이루도록 보장합니다. 블록 헤더를 제공하는 Oracle [7]과 전술한 트랜잭션과 관련된 증거를 제공하는 릴레이어라는 두 개의 독립된 엔티티를 결합하여 이를 달성합니다.

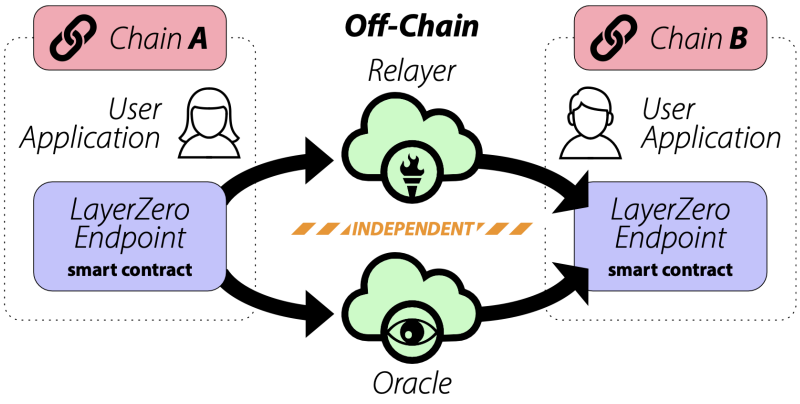

Figure 2: LayerZero ensures the validity of cross-chain communication by requiring that two independent entities, the Oracle and Relayer, corroborate the transaction.

Layer Zero에 대한 인터페이스는 Lightweight on-chain 클라이언트이며 이를 Layer Zero 엔드포인트라고 부릅니다. 각(지원 대상) 체인에는 1개의 레이어 제로 엔드 포인트가 있으며, 레이어 제로 엔드 포인트가 있는 체인은 레이어 제로 엔드 포인트가 있는 다른 체인과 관련된 크로스 체인 트랜잭션을 실행할 수 있습니다. 기본적으로는 모든 노드가 다른 모든 노드에 직접 접속하는 완전 접속 네트워크가 구축됩니다. minor boilerplate code에서는 모든 블록체인이 지원됩니다. 섹션 5에서는 Etherinum 블록체인에 LayerZero를 구현하는 사례 연구를 통해 이 과정을 시연합니다.

네트워크상의 다른 체인과 직접 크로스 체인트랜잭션을 실행할 수 있게 되면 Cross-chain decentralized

Exchange, Multichain yield aggregator, Cross-chain lending 등 이전에는 실행할 수 없었던 대규모 어플리케이션 클래스에 기회가 열립니다.

섹션 6에서는 이러한 애플리케이션 몇 가지에 대해 자세히 설명합니다. Layer Zero를 통해 사용자는 체인 간에 자유롭게 유동성을 이동할 수 있으며, 단일 유동성 풀이 서드파티 시스템이나 중간 토큰을 거치지 않고도 다양한 체인 및 에코시스템에 걸친 여러 분산형 금융(DeFi) 애플리케이션에 참여할 수 있습니다.

2. Background

Layer Zero의 토대를 마련하기 위해, 우리는 관련된 기존 시스템을 검토하여 새로운 애플리케이션의 요구를 충족시키지 못하는 이유를 설명합니다. 논의는 Layer Zero 위에 Cross-chain Exchange을 구축하는 방법에 대한 자세한 설명으로 마무리됩니다.

2.1 관련 작업

이 섹션에서는 1) Cross-chain interaction space에서 중요한 플레이어, 2) Trustless valid delivery의 이상에 미달하는 이유 3) Layer Zero가 어떻게 그 격차를 좁히는지에 대한 이해를 쌓습니다.

Ethernet [8]은 스마트 컨택을 통해 구축된 탈중앙 금융 애플리케이션을 위한 가장 인기 있는 플랫폼입니다. Ethereum은 Turing-complete 프로그래밍 언어로 기반 블록체인을 확장하여 디스트리뷰티드 애플리케이션 라이브러리가 개발자 친화적인 추상화를 통해 기반 체인의 강력한 보안 속성을 활용할 수 있도록 합니다. 그러나 기본 블록체인의 낮은 트랜잭션 속도는 초당 약 15-45 트랜잭션[9]으로 인해 Ethernet 블록체인을 직접 실행하도록 구축된 애플리케이션의 인기를 제한하는 심각한 확장성 병목 현상이 입증되었습니다.

프로그래밍 모델과 인기 때문에 많은 inter-chain communication 기법은 서드파티 체인과 Ethereum의 인터페이스를 중심으로 이루어집니다. 레이어제로(LayerZero)는 중간자 없이 바로 Ethereum과 상태를 주고받을 수 있는 기능을 제공하므로 사용자와 애플리케이션은 아래 설명된 솔루션의 비용 및 병목 현상 없이 Ethereum 체인의 안정성과 신뢰성을 활용할 수 있습니다.

Ethereum 2.0 [22]은(는) Ethereum의 확장성, 보안 및 지속가능성 단점을 해결하기 위해 제안된 업그레이드 세트입니다. Ethernet 2.0은 과부하된 Ethernet 메인 체인에 모든 트랜잭션을 집중시키는 대신 부하를 분산하는 샤드 체인을 도입습니다. 작업 증명에서 지분 증명으로 전환하면 51%의 공격 가능성을 제거하고 트랜잭션당 에너지를 절감할 수 있습니다. 이러한 진보는 Layer Zero와 거의 직교한다.

단, Ethereum의 인기를 끌어올려 편리하고 저렴한 체인 간 통신에 대한 수요를 만든다는 점은 제외합니다.

Polygon [17](이전의 Matic Network)는 Ethereum의 throughput과 주권 문제를 해결하는 레이어 2 네트워크입니다. Ethereum은 블록체인 개발을 위한 가장 인기있는 플랫폼임에도 불구하고 낮은 처리량[10]에 시달리고 있어 특정 애플리케이션에는 적합하지 않습니다.

Polygon은 개별 체인의 확장성과 독립성을 Ethereum의 커뮤니티 및 보안과 결합한 애플리케이션별 Ethernet 호환 사이드체인을 제공합니다. 특수 애플리케이션 또는 스루풋 집약적인 애플리케이션은 사이드 체인에서 실행되며 메인 Ethernet 체인으로 정기적으로 통합됩니다.

이와는 대조적으로 Layer Zero는 체인 간 직접 통신을 가능하게 하는 하위 레벨의 플랫폼이며, 폴리곤 프로토콜의 복잡함 없이 Ethernet 체인으로의 전송을 용이하게 하기 위해 사용됩니다.

Polkadot[26]은 개방된 cross-chain ecosystem 생태계의 가능성을 보여주는 초기 사례입니다. Polkadot에서는 도메인 고유의 많은 병렬 체인("패러체인")이 공통 릴레이 체인을 통해 연결되어 토큰과 데이터가 서로 흐를 수 있습니다. 단, 체인간 통신은 항상 이 릴레이 체인을 통과하기 때문에 추가 비용이 발생합니다. Layer Zero는 Polkadot과 동일한 로우레벨 통신 플랫폼을 제공하며, 온체인 미들맨에 의해 필요한 추가 트랜잭션을 수반하지 않습니다.

TORChain [23]은 서드파티 체인 간에 토큰을 전송하기 위해 쌍으로 구성된 liquidity pool을 사용하는 DEX입니다. 각 유동성 풀은 특정 서드파티 통화를 공통 교환 매체 역할을 하는 RUNE라는 이름의 TORCain 네이티브 토큰에 결합합니다.

이 공통 매체가 없다면 모든 통화 쌍에는 유동성 풀이 필요하며, 이는 풀 수가 통화 수의 제곱에 따라 확장된다는 것을 의미합니다.

유감스럽게도 RUNE는 이 scalability 문제를 해결하지만 트랜잭션 프로세스에서는 대량의 오버헤드가 발생하기 때문에 심플한 조작이 매우 복잡해집니다. 이것은 TORChain 트랜잭션 algorithm의 복잡성에서 명백합니다. ayerZero는 TORChain 고유의 확장성 보틀넥, 번거로운 중간 통화 또는 중량 프로토콜 없이 직접 체인 간 통신을 제공합니다.nySwap [2]는 TORChain과 마찬가지로 쌍으로 토큰을 쉽게 교환할 수 있도록 설계된 DEX입니다.

AnySwap은 Fusion distributed control rights management를 기반으로 intermediate 토큰인 ANY를 필요로합니다.[6]. TORCain과 마찬가지로 ANY 중간 토큰을 사용하면 불필요한 오버헤드, 지연 및 추가 전송 요금이 발생합니다.

Cosmos[5]는 지원되는 체인 간에 임의의 메시지를 전송할 수 있는 블록 체인 네트워크 기술입니다. 코스모스는 Cosmos Hub에 구축된 체인 간의 메시지를 용이하게 하기 위해 Tendermint BFT [21]에 구축된 IBC [14] 프로토콜이 포함되어 있습니다.

코스모스는 레이어 제로와는 2가지 점에서 다릅니다. (1) IBC는 완전한 온체인 라이트노드를 실행하고 (2) IBC는 빠른 파이널리티 [24]체인 간 직접 통신만 제공합니다. IBC의 이러한 제한과 함께 컨센서스를 용이하게 하기 위한 intermediate 체인의 사용은 레이어 제로 등의 일반적인 통신 레이어가 아닌 Anyswap, TORChain 또는 Polkadot과 유사합니다. Cosmos는 또한 Gravity Bridge [12]라고 불리는 Anyswap 또는 TORChain과 유사한 특성을 가진 DEX를 제공합니다. Cosmos나 IBC와 달리 LayerZero는 Trustless 옴니체인 메시징을 제공하며 Ethernet이나 Bitcoin과 같은 확률론적 최종성을 제공하는 체인을 포함하여 모든 체인에서 실행되도록 확장할 수 있습니다.

Chainlink [7, 4]는 분산형 오라클 네트워크(DON)를 구축하고 연결하기 위한 프레임워크입니다. 스마트 계약은 조작이 불가능하지만, 온 체인의 특성으로 인해 폭넓은 채택에 필수적인 기본적인 연결이 어렵습니다.

즉, 스마트 계약은 주가, IoT 장치 측정, 안전한 오프 체인 연산 출력 등 계약 실행에 필요한 오프 체인 데이터를 가져올 수 없습니다. DON은 스마트 계약의 변조 방지 속성을 스마트계약이 의존하는 데이터 소스 및 외부 리소스로 확장합니다(중앙 entity에 대한 신뢰 없이).

DON에서 사용자의 스마트 계약은 Chainlink 인터페이스 스마트 계약에 온체인 요청을 하고, Chainlink 인터페이스는 여러 개의 개별 Oracle 노드에 이벤트를 게시합니다. 각 Oracle 노드는 요청된 정보에 대해 여러 데이터 소스를 쿼리하고 이를 집계하여 오류 또는 악의적인 소스를 필터링하고 선택적으로 데이터에 대해 신뢰 최소화 계산을 수행합니다.

Oracle 노드는 Chainlink 인터페이스 계약에 응답합니다. Chainlink 인터페이스 계약에서는 2nd level of

aggregation 을 수행하여 오류 또는 악의적인 Oracle을 필터링합니다.

이 이중 계층 필터링은 개별 오라클 또는 데이터 소스에 대한 신뢰 없이 최종 데이터에 대한 신뢰를 보장합니다. 그 결과, 체인 링크는 견고한 정보 취득 네트워크와 업계 전체에서 널리 사용되고 있는 안전한 오프체인 계산 솔루션을 제공합니다. Chainlink DON 프레임워크를 활용함으로써 Layer Zero 프로토콜은 서로 다른 체인 간에 신뢰할 수 없는 메시지 전달을 보장하는 기능을 확보합니다.

2.2 실제 Layer Zero

개발자는 Layer Zero를 사용하여 복잡한 크로스 체인 애플리케이션을 구축할 수 있습니다. 신뢰성을 희생하거나 복잡한 중간 체인/스마트 계약을 체결하지 않아도 됩니다. 그림 3은 Exchange를 구축하는 경우의 레이어 제로 기능을 나타내고 있습니다.

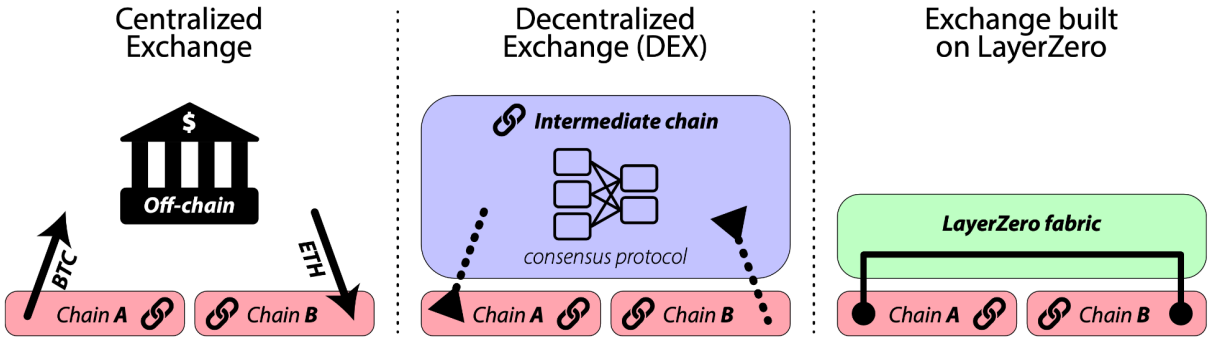

Figure 3: LayerZero is a building block for cross-chain applications. This figure visualizes the architectural differences between a centralized exchange, a decentralized exchange, and a cross-chain bridge built using LayerZero as its underlying communication primitive.

왼쪽에 표시된 CEX에서는 사용자가 토큰을 중앙의 신뢰할 수 있는 주체에게 맡길 필요가 있습니다. 해당 주체는 그 예금을 오프체인으로 추적하여 사용자가 요청하면 다른 체인으로 coin을 부여합니다. 이 권한을 신뢰하는 것은 블록체인의 사용 목적에 어긋나는 것입니다.

DEX는 바로 그 지점에서 출발했습니다. 가운데 이미지는 일반적인 분산형 거래소가 스마트 계약 방식의 합의 프로토콜을 사용하여 체인 B에서 동전을 자동으로 발행하는 모습을 대략적으로 보여줍니다. 이러한 방식으로 DEX는 중앙화 된 신뢰할 수 있는 오프 체인 미들맨 없이도 교환이 가능하다는 사실을 보여주고 있습니다.

그러나 한 가지 중요한 제한사항은 DEX가 intermediate 토큰과 intermediate 체인을 포함하며 사용자가 원하는 실제 토큰과 달리 체인 B에 intermediary 또는 wrapped 토큰만 발행한다는 것입니다. 그런 다음 사용자는 추가 트랜잭션에서 중간 토큰(예: RUNE) 또는 래핑된 토큰(예: ANY)을 원하는 토큰으로 교환해야 합니다. 이 intermediary/wrapped 토큰, 두 번째 트랜잭션 및 intermediate 체인은 모두 single seamless Transaction이어야 하는 불필요한 오버헤드입니다.

그림 3의 오른쪽은 Layer Zero에 구축된 거래소의 모습을 나타내고 있습니다.체인 A는 1)체인 A에서 local 트랜잭션을 용이하게 하고 2) 체인 B에서 사용자에게 안전하게 토큰을 부여할 수 있음을 어플리케이션에 통지하는 Single cross-chain transaction을 시작할 수 있습니다.

이 application에서는 Layer Zero를 사용하면 중간 토큰을 포함하지 않는 clean and minimal single-transaction swap이 가능합니다. 실제 exchange protocol은 cross-chain transaction의 양쪽에 있는 스마트 계약에 의해 처리되며, 레이어 제로(Layer Zero)는 둘 사이의 메시지를 전달합니다. 이는 상당한

유연성을 제공하며, source 및 destination 체인의 스마트 계약에 의해 처리되는 대부분의 high level exchange logic과 엔드 투 엔드 원칙[18]을 따릅니다.

3. Valid Delivery

이 섹션에서는 trustless inter-chain communication의 기본 특성에 대해 설명합니다. 상이한 체인상의 트랜잭션 검증의 문제를 정식으로 다루기 위해서, Valid delivery의 개념을 정의합니다. Valid delivery은 다음 두 가지 guarantees를 제공함으로써 Cross-chain token transfer을 가능하게 하는 통신 프리미티브입니다.

1. 네트워크를 통해 송신되는 각 메시지 m은 sender-side chain상의 트랜잭션 t와 결합된다.

2. 메시지 m은 관련 트랜잭션 t가 유효하고 sender-side chain 상에서 커밋된 경우에만 수신자에게 전달된다.

중앙거래소는 클라이언트와 거래소의 합의를 통해 하나의 체인에서 거래소로 토큰을 전송하고, 거래소는 그 토큰을 수신하면 약간의 balance(non-cryptocurrency)을 발행한다는 점에서 valid delivery을 보증합니다. 이 non-cryptocurrency balance은 이용 가능한 모든 체인에서 인출할 수 있으며, 이는 지원되는 각 체인에서 거래소에 의해 유지되는 광범위한 유동성 풀에 의해 실현됩니다. 거래소는 이 거래의 미들맨 역할을 하며, 사용자는 거래의 목적을 위해 거래소를 신뢰해야 합니다. 단, 악의적인 교환 또는 침해된 교환은 클라이언트로부터 토큰을 받아 잔액을 발행한 후 다른 체인에서 잔액을 인출하는 것을 거부하여 실질적으로 사용자로부터 토큰을 훔칠 수 있습니다. 사용자가 거래소를 신뢰할 의향이 있다고 해도, 최근 몇 년 동안 암호화폐 거래소를 해킹하거나 해킹하려는 시도가 많으므로 [15] 신뢰할 수 있는 중개인을 필요로 하지 않는 솔루션이 사용자에게 더 잘 제공됩니다. 더 높은 수준에서, 암호 화폐의 핵심 세입자(-->의역필요) 중 하나는 은행과 같은 중앙집권화된 실체로부터의 독립성이기 때문에, 중앙집권화된 거래소에 다시 의존하는 것은 그들의 목적을 저버리는 것입니다.

중앙 집중식 교환을 사용하는 대신 TORCain [23] 또는 AnySwap [2]와 같은 분산형 교환을 사용할 수 있습니다. 기존의 모든 DEX는 앞서 말한 transaction t를 위해서 TORChain의 경우 RUNE, AnySwap의 경우 ANY와 같은 intermediate token을 사용합니다. 이러한 중간 토큰은 각 DEX의 재특정 프로토콜에 의해 관리되기 때문에, 악의적인 사용자가 중간 토큰을 위조하는 것이 불가능하기 때문에, DEX는 유효한 전달을 요구할 수있습니다. 기존 DEX 솔루션은 송신자의 토큰을 기간 내 토큰으로 변환하는 것과 중간 수신자를 수신자 체인에서 원하는 "실제" 토큰으로 변환하는 두 가지 중간 트랜잭션을 수반하기 때문에 이상적이지 않습니다. 이것에 가세해, 유저는, 소스 체인상의 트랜잭션을 확인하고, 동사의 제조 의향을 전달하는 중간 합의 레이어를 완전하게 신뢰하는 것이 필요합니다. 기존 교환에서는 크로스 체인 토큰 전송이 가능하지만 불필요한 복잡성과 비용을 감수해야 합니다.

이것의 단점은 크로스 체인 애플리케이션이 광범위하게 채택되지 않았다는 데 있습니다. 체인 간 트랜잭션 문제에 대한 이상적인 해결책은 신뢰할 수 있는 중간 엔티티를 포함하지 않고 체인 간에 단일 원스왑 트랜잭션을 사용하는 것입니다.즉, 신뢰할 수 없는 유효한 디리버리입니다. 당사의 작업에서는 토큰뿐만 아니라 임의의 사용자 데이터의 신뢰할 수 없는 유효한 전달을 제공하는 범용 메시징 프로토콜을 구현하고 있습니다. 분산 교환 또는 기타 DeFi 애플리케이션은 크로스 체인 트랜잭션을 제공하기 위해 우리의 메시싱 프리미티브를 사용하여 구현될 것이며, low-level 메시징 프로토콜에 의해 제공되는 유연성은 더 높은 수준의 애플리케이션이 이전에는 불가능했던 광범위한 기능을 구현할 수 있게 합니다.

4. Design

Layer Zero의 핵심은 trustless valid delivery을 제공하는 통신 프로토콜입니다. 프로토콜은 섹션 4.1에 소개된 일련의 구성 요소를 기반으로 구축되었습니다. 섹션 4.2에서 전송 프로토콜의 통신 흐름에 대해 논의하고, 섹션 4.3에서 LayerZero가 신뢰할 수 있는 중개 서비스를 포함하지 않고 유효한 전달을 달성할 수 있는 방법을 설명하고, 섹션 4.4에서 저비용 스마트 계약 기반 라이트 클라이언트 엔드포인트 설계를 제시합니다

4.1 Layer Zero 컴포넌트

Layer Zero 엔드포인트는 Layer Zero에 대한 사용자 방향 인터페이스입니다. 레이어 제로 네트워크의 각 체인에는 일련의 온체인 스마트 계약으로 구현된1개의 레이어 제로 엔드 포인트가 있습니다. Endpoint의 목적은 사용자가 LayerZero 프로토콜 백엔드를 사용하여 메시지를 보낼 수 있도록 하는 것입니다.

Layer Zero Endpoint는 다음 4개의 모듈로 분할됩니다.

1) Communicator
2) Validator
3) Network
4) Libraries

Communicator, Validator 및 네트워크 모듈은 엔드 포인트의 핵심 기능을 통합하며(그림 4), Layer Zero에서 지원되는 각 새로운 체인은 추가 라이브러리로 추가됩니다. 이 설계에서는 3개의 코어 모듈을 변경하지 않고 새로운 체인에 대한 지원을 추가할 수 있습니다. 섹션 4.4에서 각 모듈의 기능을 설명합니다.

Oracle은 다른 LayerZero 컴포넌트와는 독립적으로 한 체인의 블록 헤더를 읽어 다른 체인으로 전송하는 메커니즘을 제공하는 서드파티 서비스입니다. 이론적으로 이 Oracle은 이 메커니즘을 제공하는 모든 서드파티 서비스가 될 수 있지만 실제로는 분산형 Oracle 네트워크의 현재 업계 선두 업체인 Chainlink [7, 4]를 사용할 것으로 예상됩니다.

릴레이어는 Oracle과 기능이 유사한 오프 체인 서비스이지만 블록 헤더를 가져오는 대신 지정된 트랜잭션의 증명을 가져옵니다.

유효한 전달을 보증하기 위해서는 LayerZero protocol을 사용하여 전송되는 특정 메시지에 대해 Oracle과 Relayer가 서로 독립되어 있어야 합니다. 프로토콜 자체는 릴레이어의 특정한 구현을 요구하지 않으며, 이론적으로 LayerZero의 사용자들은 그들 자신의 릴레이어 서비스도 구현할 수 있습니다. 이 설계를 통해 사용자는 릴레이어가 Oracle과 결탁할 수 없다는 것을 확신할 수 있으며, 이러한 독립성을 통해 섹션 4.3에서 보듯이 신뢰할 수 없는 검증된 델리베리를 구현할 수 있습니다. 실제로 Layer Zero는 릴레이어 서비스를 프로바이더하고 Oracle은 Chainlink의 분산형 오라클 네트워크 및 관련 합의 메커니즘에 의해 처리됩니다.

4.2 Layer Zero 프로토콜

그림 4는 단일 Layer Zero 메시지의 유효한 전달에 관련된 절차를 나타내고 있습니다. 그림에서 동그라미로 둘러싸인 각 숫자는 프로토콜의 단계를 나타내며 이 섹션의 단락과 일치합니다. 여기에서는 체인 A의 사용자 어플리케이션이 Layer Zero를 통해 체인 B의 사용자 어플리케이션에 단일 메시지를 보내는 예를 설명합니다. 섹션 5에서는 두 개의 Etherinum Vir-tual Vachines 간에 메시지를 보내는 경우 가변 컴포넌트와 프로토콜 단계가 어떻게 구현되는지 설명합니다.

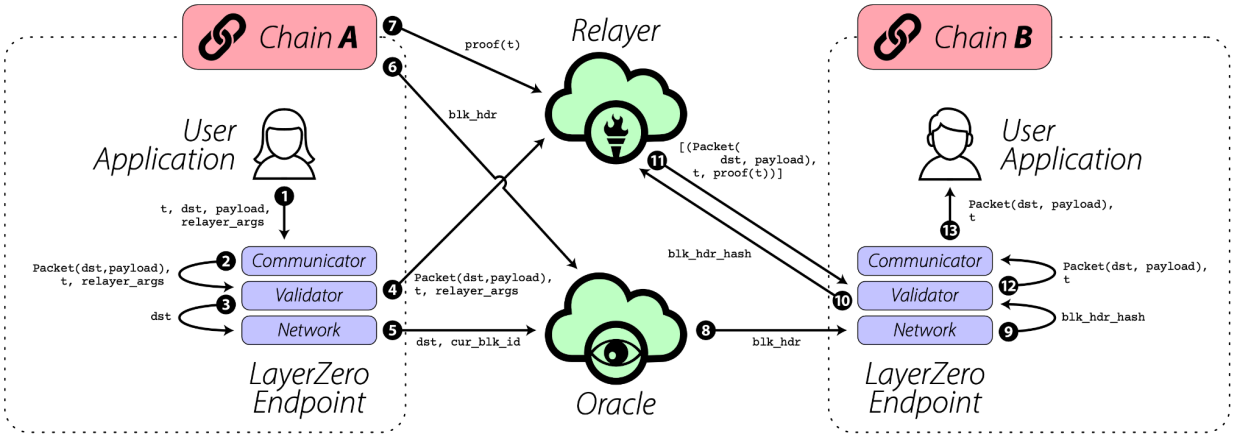

Figure 4: The communication flow in a single LayerZero cross-chain transaction.

스텝 1: 체인 A(App A)의 사용자 어플리케이션은 일련의 액션을 트랜잭션T의 일부로 간주합니다.트랜잭션 식별자 t에 의해 트랜잭션T를 특정해서 식별합니다.이 식별자의 형식은 체인A의 유형에 따라 다를 수 있습니다. 트랜잭션T 에 포함되는 스텝은, T 에 조건 붙여진 유효한 전달을 가지는 LayerZero 를 개입시켜 메세지를 송신하는 것입니다.

범용성을 잃지 않는 선에서 이 시나리오에서는 다음과 같이 가정합니다. App A는 우리의 reference 릴레이어를 사용하고 있습니다. App A는 다음 정보를 포함하는 요청을 LayerZero Communicator로 보냅니다.

- t: T의 고유 트랜잭션 ID(체인의 유형에 따라 다를수있음)
- dst: 체인 B의 스마트 계약을 나타내는 글로벌 ID.
- 페이로드: 앱 A가 앱 B로 전송하고자 하는 모든 데이터.
- 릴레이 arg: 앱 A가 참조 릴레이어를 사용하려는 경우 지불 정보를 설명하는 인수입니다

스텝 2: Communicator는 dst 및 payload를 포함하는 Layer Zero 패킷을 구축하여 Packet(dst, payload)이라고 하며 t 및 relayer arg와 함께 Validator로 전송합니다.

스텝 3: 검증자는 t와 dst를 네트워크에 송신합니다. 이 스텝은 체인A의 현재 블록의 블록헤더를 체인B로 송신할 필요가 있음을 네트워크에 통지합니다.

스텝 4(?): Validator는 Packet(dst, payload), t 및 relayer arg를 Relayer에 전송하여 T의 transaction proof를 프리페치(임시저장, 미리보기, 캐시역할)하여 최종적으로 체인B로 송신할 필요가 있음을 Relayer에 통지합니다. 이것은 스텝 3과 동시에 발생합니다.

스텝 5: 네트워크는 dst와 현재 트랜잭션의 블록 ID(cur blk id)를 Oracle로 전송합니다. 이것은 Oracle이 체인 A의 현재 블록의 블록 헤더를 가져와 체인 B로 전송하도록 알려줍니다. 같은 블록 내에서 여러 Layer Zero 트랜잭션이 발생한 경우 스텝5는 1회만 실행됩니다.

6단계: Oracle은 체인 A에서 블록 헤더(blk hdr)를 읽습니다.

스텝 7(?): 릴레이어는 체인 A에서 트랜잭션 T(proof(t))와 관련된 트랜잭션 증빙을 읽고 오프 체인으로 보존됩니다. 스텝 6과 7은 서로 비동기적으로 실행됩니다

스텝 8: Oracle은 blk hdr에 대한 블록 코어가 체인 A에서 안정적으로 커밋된 것을 확인한 후 blk hdr을 체인 B의 네트워크로 전송합니다. 이러한 상황이 언제 발생할지를 결정하는 메카니즘은 체인마다 다르지만 일반적으로 블록 확인을 몇 번 기다리는 것이 포함됩니다.

스텝 9(?): 체인 B의 네트워크는 블록해시를 blk hdr 해시로 지정한 후 검증자에게 전송합니다.

순서 10: 검증자는 blk hdr 해시를 릴레이어로 전송합니다.

스텝 11: blk hdr 해시를 수신하면 Re-layer는 현재 블록과 일치하는 모든 Packet(dst, payload), t, proof(t) 튜플 목록을 전송합니다. 여러 사용자가 동일한 엔드포인트 간에 동시에 메시지를 보내는 경우 동일한 블록 내에 여러 개의 패킷 및 관련 트랜잭션 증명이 있을 수 있습니다.

순서 12: 검증자는 수신된 트랜잭션 증명과 네트워크에 의해 저장된 블록헤더를 사용하여 관련 트랜잭션T가 유효하고 커밋되었는지 여부를 검증합니다. 블록 헤더와 트랜스액션프루프가 일치하지 않으면 메시지는 폐기됩니다. 일치하면 패킷(dst, payload)이 Communicator 에 송신됩니다.

순서 13: Communicator는 App B에 패킷(dst, payload)을 송신합니다.

4.3 Trustless Valid delivery 달성

Trustless: Layer Zero 설계의 핵심은 사용자가 Lay-er Zero의 컴포넌트를 신뢰할 필요가 없다는 것입니다. 신뢰(강력한 결점)를 요구하는 대신 Oracle과 Relayer 사이에 약한 조건의 independence만 있으면 됩니다. 이 신뢰 대신 독립성을 요구하는 것은 레이어 제로(Layer Zero)가 효율적이고 가볍다는 것의 한 측면입니다. Oracle과 Relayer 사이에 악의적인 담합이 없는 한 Layer Zero는 유효한 전달을 보증합니다.

Valid delivery : 섹션 4.2의 LayerZero 프로토콜에 의해 메시지 m은 m과 관련된 트랜잭션 t의 트랜잭션 증빙이 스텝 12에서 Validate될 수 있는 경우에만 Communicator에 의해 사용자 애플리케이션에 전달됩니다. 이 검증 순서는 블록헤더와 트랜잭션 증명이 일치하는 경우에만 성공합니다.이 검증 순서는 다음 두 가지 시나리오에서만 발생합니다.

1. Oracle이 제공하는 블록 헤더와 릴레이러가 제공하는 트랜잭션 증명은 모두 유효합니다.
2. Oracle에 의해 제공된 블록 헤더와 릴레이어에 의해 제공된 트랜잭션 증명은 모두 유효하지 않지만 여전히 일치합니다.

시나리오 2는 Oracle과 Re-layer가 결탁한 경우에만 발생할 수 있습니다.이는 특정 블록 헤더를 인식하지 않고 블록 헤더에 대해 검증할 수 있는 트랜잭션 증빙을 전송하는 것이 통계적으로 불가능하기 때문입니다. 단, Layer Zero의 설계에서는 섹션 1에서 설명한 바와 같이 담합의 가능성이 배제됩니다. 따라서 메시지가 수신측 사용자 응용 프로그램에 전달될 경우 유효한 전달 속성을 충족할 수 있습니다.

제3절에서 설명한 바와 같이, 중간 엔티티나 토큰을 신뢰하지 않고 신뢰할 수 없는 유효한 전달, 즉 유효한 전달을 보장할 수 있는 통신 프로토콜이 크로스 체인 트랜잭션을 가능하게 하는 이상적인 솔루션입니다. Layer Zero는 이 속성을 증명한 최초의 유일한 시스템입니다. 이 사실은 사용자가 선호하는 크로스 체인메시징 방법으로서 레이어 제로(Layer Zero)를 채택하도록 유도합니다.

4.4 Layer Zero 엔드 포인트

Layer Zero Endpoint는 현재 Lay-erZero 네트워크에 포함된 각 체인에 스마트 계약의 시리즈로 구현되어 있습니다. Layer Zero Endpoint의 핵심 기능은 Commu-nection, Validation 및 Network의 3가지 모듈로 캡슐화되어 있습니다. 이러한 모듈은, 송신측(Communicator to Validator to Network)의 스택에 메세지가 송신되어 수신측 스택에 업 하는 등, 네트워크 스택과 같이 동작합니다.

코어 모듈 외에 Layer Zero Endpoint는 라이브러리를 통해 확장할 수 있습니다.Libraries는 특정 체인의 통신 처리 방법을 정의하는 보조 스마트 계약입니다. LayerZero 네트워크의 각 체인에는 연결된 라이브러리가 있으며 각 Endpoint에는 모든 라이브러리의 복사본이 포함됩니다. 이 모듈러 설계를 통해 Layer Zero 네트워크를 빠르고 쉽게 확장하여 새로운 체인을 온 디맨드로 추가할 수 있습니다. 또, 2개의 체인간의 통신에서는, 각각의 라이브러리가 양끝에 존재하는 것만으로, Layer Zero 는, 임의의 노드 쌍간의 트랜잭션을 조정할 수 있는 완전 접속 네트워크로 할 수 있습니다.

4.5 Layer Zero Endpoint 비용 확장성

많은 독자들이 지적하고 있듯이, 레이어 1 체인에서 스마트 콘택트를 실행하는 것은 특히 저장 데이터의 양이 증가함에 따라 비용이 많이 들 수 있습니다. Layer Zero Endpoint를 실용화하기 위해서는 가능한 한 경량 클라이언트를 설계해야 했습니다. Golden Gate [11]와 같은 SMR (크로스 체인 스테이트 머신 레플리케이션)을 통한 신뢰할 수 없는 크로스 체인 검증에 대한 이전 작업은 Ethermin과 같은 인기 레이어 1 체인을 실행하는 데 하루에 수백만 달러가 소요될 수 있습니다

Ethernet의 경우 15가 되는 블록 확인을 몇 번 들은 후 특정 블록 헤더의 체인이 됩니다. 정확히 말하면, Lay-erZero 프로토콜(섹션 4.2)의 8단계는 Oracle이 체인 A에서 15개의 블록 확인을 수신한 후에만 실행됩니다.

Layer Zero 엔드포인트: Layer Zero Endpoint는 섹션 4.4에서 설명하는 4개의 주요 모듈로 구성된 일련의 스마트 계약으로 구현됩니다. Etherinum 블록체인을 포함한 대부분의 기존 블록체인에 대해 Communica-를 구현할 수 있습니다.

그림 5: EVM 엔드포인트 레이어 제로 패킷 레이아웃

이 문제를 해결하기 위해 우리는 가능한 한 가장 가벼운 클라이언트를 설계하기 시작했습니다. 우리의 주요 견해는 클라이언트 내에서 블록 헤더를 복제하고 저장할 필요가 없다는 것입니다. 대신, 우리는 필요한 크로스 체인 헤더와 트랜잭션 증거를 오프 체인 엔티티인 Oracle과 Relayer에게 가져오는 작업을 위임합니다. 그 결과, LayerZero Endpoints는 놀라울 정도로 가벼워져 Etherin과 같이 비용이 많이 드는 [20] 체인에서도 비용 효율이 높습니다.

5. 도입 사례: EVM에서의 Layer Zero

이 섹션에서는 EVM(Ethernet Virtual Machine)에서 LayerZero를 실행하기 위한 지원을 구현하는 방법에 대해 간략히 설명합니다[13]. 간결성을 위해, 우리는 체인에 따라 IM-plementation이 다를 가능성이 있는 시스템의 측면에 초점을 맞추고 우리의 구현이 어떻게 Ethernet 체인의 특정 요구사항을 처리하는지 강조한다. 섹션 4.1에서 설명한 바와 같이 현재 버전의 Layer Zero는 Oracle 서비스를 제공하기 위해 Chainlink에 의존하며 사용자가 당사가 제공하는 Re-layer 서비스를 사용할 것으로 예상됩니다.

Layer Zero 패킷: 레이어 제로 패킷의 형식은, 송신원체인과 행선지 체인에 의해서 다릅니다. 그림 5에서는 EVM 엔드포인트용 Layer Zero 패킷의 정확한 레이아웃[19]을 나타냅니다. 각 필드는 다음과 같이 기능합니다.

송신측 체인 트랜잭션 안정성: 메시지 트랜잭션이 소스 체인에서 안정되도록 하기 위해 분산형 ora-cle 네트워크의 고유 속성에 의존합니다.Oracle은 수신처에 대해서만 통지합니다.

tor, Validator 및 Network를 각각 개별 스마트 연락처로 지정합니다. 단, 이 설계에서는 요건이 다른 (향후) 체인상의 레이어 제로 엔드포인트 도입이 배제되는 것은 아닙니다.

LayerZero Endpoint의 Library 컴포넌트는 이 사례연구에서 Etherinum 블록체인을 지원하기 위한 핵심 요소입니다. 그림 5와 같은 EVM 고유의 LayerZero 패킷의 구성을 처리하고 EVM 스마트 계약 주소 정보의 인코딩 및 디코딩을 처리하기 위해 라이브러리를 구현합니다.

도서관의 추가적인 책임은 거래 증거를 검증하는 것과 관련된 실제 계산을 처리하는 것이다. 당사의 EVM 라이브러리는 EVM 블록 상의 트랜잭션에 대한 Merkle-Patricia Tree 검증[16]을 처리하며, Golden Gate [11]에 의한 오픈 소스 구현을 기반으로 합니다.

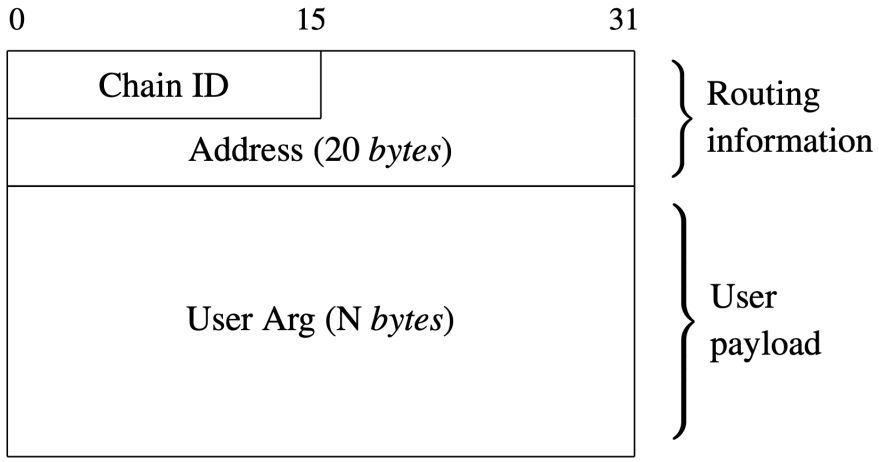

Figure 5: LayerZero packet layout for EVM endpoints.

6. Layer Zero 상의 응용 프로그램

크로스 체인 분산형 교환: 섹션 2.2에서 간단히 설명한 바와 같이 Layer Zero는 na-tive 자산만을 취급하는 크로스 체인 DEX(크로스 체인 브리지)를 가능하게 합니다. 래핑된 토큰을 발행하거나 중간 사이드체인을 통과하는 기존 DEX 설계와는 달리, 체인 간에 메시지를 보내기 위해 Layer Zero를 사용하여 구축된 DEX는 두 체인 모두에 유동성 풀이 존재하도록 구축될 수 있으며, 사용자는 단순히 한 풀에 자신의 네이티브 자산을 저장하고 다른 풀에서 네이티브 자산을 인출할 수 있습니다. Lay-erZero의 메시징 프리미티브는 다이렉트 브리지(1:1 가격 설정), 자동화된 마켓 메이킹(ab = k 가격 설정) 및 기타 파생 프로그램(예: Curve DAO 가격 설정[25])을 활성화할 수 있을 만큼 강력합니다. Layer Zero가 제공하는 유효한 전달을 보증함으로써 광범위한 분산형 Exchange 애플리케이션을 사용할 수 있습니다.

멀티 체인 수율 집계기: 현재의 수익률 집계는 일반적으로 Sin-Gle-Chain 생태계의 범위 내에서 운영되며 Yearn Fi-Nance[27]와 같은 프로젝트는 단일 체인 전략을 사용하여 수익률 집계를 가능하게 합니다. 이러한 단일 체인 수익률 집계 시스템의 주요 약점 중 하나는 현재의 에코 시스템 이외의 수익률 기회를 활용할 수 없다는 것입니다. 따라서 최고의 수익률도 상당 부분 놓칠 수 있습니다. 크로스 체인 트랜잭션에 레이어 제로(Layer Zero)를 사용하는 수율 어그리게이터는, 이 어그리게이터에 의해서, 이 어그리게이터를 이용하는 스트래티지를 가능하게 됩니다.

모든 에코시스템에 걸쳐 최고의 기회를 제공하여 높은 수익률의 기회에 대한 접근성을 높이고 사용자가 시장의 비효율성을 활용할 수 있도록 지원합니다. 멀티체인 수율집약기는 단일 체인 수율집약기보다 훨씬 우수합니다. 최악의 경우 전략은 하나의 체인에서만 기회를 활용하는 것으로 저하되며, 최선의 경우 선택할 수 있는 기회가 매우 많아지기 때문입니다.

멀티 체인 대여: 오늘날 사용자는 자산을 보유하지 않는 체인(chain)에서 기회를 활용할 수 있는 쉬운 방법이 없습니다. 예를 들어 ETH에 통합된 자산을 가진 사용자가 Polygon [17]에 대한 기회를 활용하려고 한다고 가정합니다. 선택사항은 (1) 자산 베이스 전체를 다른 체인으로 이동하여 원하는 통화로 변환하거나 (2) Ethermin에서 자산을 빌려 원하는 자산을 빌린 다음 해당 자산을 대상 체인에 연결하는 것입니다. Layer Zero는 사용자가 모든 자산 기반을 Etherium에 두고 대출한 다음 폴리곤의 MATIC에서 직접 빌릴 수 있는 대출 프로토콜을 가능하게 한다. 이를 통해 브리지 및 스왑 수수료와 같은 중간 비용이 제거됩니다.

이들 3가지 예는 Layer Zero가 실현하는 많은 가능성 중 극히 일부에 불과합니다. Layer Zero를 활용함으로써 개발자는 체인 간 트랜잭션과 체인 내 트랜잭션 간의 다른 se-mantics에 대해 걱정하지 않고 애플리케이션을 작성할 수 있으며 사용자는 체인 간 유동성을 자유롭게 이동할 수 있습니다. 신뢰할 수 없는 크로스 체인 트랜잭션의 힘을 바탕으로 커뮤니티가 전개할 새로운 크리에이티브 애플리케이션을 기대합니다.

7. 결론

이 문서에서는 중간 트랜시션을 수반하지 않는 최초의 신뢰할 수 없는 옴니체인 상호운용성 플랫폼인 LayerZero의 설계와 구현을 소개했습니다. Lay-erZero는 신뢰할 수 없는 2개의 독립된 오프 체인 엔티티인 Oracle과 Relayer를 활용함으로써 비용이 많이 드는 크로스 체인 상태의 머신 복제나 다이어리 토큰을 필요로 하지 않고 유효한 전달을 실현할 수 있음을 보여주었습니다. 당사의 프로토콜은 임의 릴레이 서비스 사용을 배제하지 않는 방식으로 설계되었으며, 릴레이어와 오라클 간에 유착이 발생하지 않도록 보장합니다. LayerZero 프로토콜은 지원되는 체인 간의 네이티브 트랜잭션을 가능하게 하며, 새로운 Layer-erZero Endpoint 설계는 모든 체인을 지원하도록 쉽게 확장할 수 있습니다. 또한 당사의 Endpoint 디자인은 Ethermin과 같은 고가의 레이어 1 체인에서 막대한 비용을 들이지 않고 실행할 수 있을 만큼 가볍습니다. 우리는 LayerZero를 통한 교차 체인 트랜잭션을 가능하게 하기 위해 Chainlink의 분산형 오라클 네트워크와 함께 참조 릴레이어 구현을 사용하여 LayerZero에서 EVM 기반 체인에 대한 지원을 구현하는 방법에 대한 사례 연구를 제시했습니다.

레이어제로(Layer Zero)는 다양한 블록체인 생태계를 연결하고 체인과 커뮤니티 간에 유동성, 데이터, 아이디어의 마찰 없는 이동을 가능하게 하는 백본이다.

References

[1] All layer 1 blockchain protocols. https://blockchain-comparison.com/blockchain- protocols/. Accessed: 2021-5-13.
[2] Anyswap dex user guide. https://anyswap- faq.readthedocs.io/en/latest/index.html. Accessed: 2021-5-13.
[3] Binance.com. https://www.binance.com/. Accessed: 2021- 5-14.
[4] BREIDENBACH, L., CACHIN, C., CHAN, B., COVENTRY, A., ELLIS, S., JUELS, A., KOUSHANFAR, F., MILLER, A., MA- GAURAN, B., MOROZ, D., NAZAROV, S., TOPLICEANU, A., TRAME`R, F., AND ZHANG, F. Chainlink 2.0: Next steps in the evolution of decentralized oracle networks. White paper, Chain- Link, 2021.
[5] What is cosmos? https://v1.cosmos.network/intro. Ac- cessed: 2021-5-15.
[6] Dcrm - fusion.org. https://www.fusion.org/tech/dcrm. Accessed: 2021-5-13.
[7] ELLIS, S., JUELS, A., AND NAZAROV, S. Chainlink: A decen- tralized oracle network. White paper, ChainLink, 2017.
[8] Ethereum. https://ethereum.org/en/. Accessed: 2021-5- 13.
[9] Ethereum 2.0 (eth2) vision. https://ethereum.org/en/ eth2/vision/. Accessed: 2021-5-13.
[10] GALAL, H. S., ELSHEIKH, M., AND YOUSSEF, A. M. An efficient micropayment channel on ethereum. In *Data Pri- vacy Management, Cryptocurrencies and Blockchain Technol- ogy*. Springer, 2019, pp. 211-218.
[11] Golden gate - trustless-bridging ethereum (evm) blockchains - part 1: Basics. https://loredanacirstea.medium.com/ golden- gate- trustless- bridging- ethereum- evm- blockchains-part- 1- basics- d016300ea0dd. Accessed: 2021-5-14.
[12] Announcing the gravity bridge. https://blog.althea.net/ gravity- bridge/. Accessed: 2021-5-15.
[13] HILDENBRANDT,E.,SAXENA,M.,RODRIGUES,N.,ZHU,X., DAIAN, P., GUTH, D., MOORE, B., PARK, D., ZHANG, Y., STEFANESCU, A., ET AL. Kevm: A complete formal semantics of the ethereum virtual machine. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)* (2018), IEEE, pp. 204- 217.
[14] Ibcoverview—cosmossdk.https://docs.cosmos.network/ master/ibc/overview.html. Accessed: 2021-5-15.
[15] LAZARENKO, A., AND AVDOSHIN, S. Financial risks of the blockchain industry: A survey of cyberattacks. In *Proceedings of the Future Technologies Conference* (2018), Springer, pp. 368- 384.
[16] LU, Z., WANG, Q., QU, G., ZHANG, H., AND LIU, Z. A blockchain-based privacy-preserving authentication scheme forvanets. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems 27*, 12 (2019), 2792-2801.

[17] Polygon: Ethereum's internet of blockchains. https:// polygon.technology/lightpaper-polygon.pdf.

[18] SALTZER, J. H., REED, D. P., AND CLARK, D. D. End-to-end arguments in system design. *ACM Transactions on Computer Systems (TOCS) 2*, 4 (1984), 277-288.
[19] Solidity types. https://docs.soliditylang.org/en/ v0.5.3/types.html. Accessed: 2021-5-14.
[20] SPAIN, M., FOLEY, S., AND GRAMOLI, V. The Impact of Ethereum Throughput and Fees on Transaction Latency During ICOs. In *International Conference on Blockchain Economics, Security and*

Protocols (Tokenomics 2019) (Dagstuhl, Germany, 2020), V. Danos, M. Herlihy, M. Potop-Butucaru, J. Prat, and S. Tucci-Piergiovanni, Eds., vol. 71 of *OpenAccess Series in In- formatics (OASIcs)*, Schloss Dagstuhl-Leibniz-Zentrum fuer In- formatik, pp. 9:1-9:15.

[21] Tendermint. https://tendermint.com/. Accessed: 2021-5- 15.

[22] The eth2 upgrades. https://ethereum.org/en/eth2/. Ac- cessed: 2021-5-13.

[23] Thorchain. https://thorchain.org. Accessed: 2021-5-13.

[24] VIRIYASITAVAT, W., DA XU, L., BI, Z., AND SAPSOMBOON, A. New blockchain-based architecture for service interoperations in internet of things. *IEEE Transactions on Computational Social Systems 6*, 4 (2019), 739-748.

[25] WARREN, W., AND BANDEALI, A. 0x: An open protocol for decentralized exchange on the ethereum blockchain. *URI: https://github. com/0xProject/whitepaper* (2017), 04-18.

[26] WOOD, G. Polkadot: Vision for a heterogeneous multi-chain framework. White paper, Polkadot, 2016.

[27] yearn.finance. https://yearn.finance/. Accessed: 2021-5- 14.