SKALE

SKALE

The Gas Free Invisible Blockchain

Secure Gas-Free Scalability for Ethereum Applications

OCT 2024

Contents

개요

이 문서와 관련된 기술들은 개발 중이며 변경될 수 있습니다.

블록체인 확장성은 이더리움이 처음 혼잡 문제를 경험한 2017년 이후로 주목의 중심이 되어왔습니다. 수십억 달러의 가치 평가를 받는 레이어-1 토큰을 사용하는 새로운 확장 기술들이 지속적으로 발표되고 있지만, 실제로는 진전이 거의 없고 혁신도 미미합니다. 이에 우리는 이러한 방법과 기술의 궁극적인 목표가 무엇인지 물어봐야 합니다. 새로운 토큰을 출시하여 과대광고를 조성하는 것인가, 아니면 개발자와 애플리케이션의 최종 사용자에게 가치를 제공하는 것인가? 현실은 산업 혁신이 하나의 궁극적인 목표를 추구해야 한다는 것입니다: 탈중앙화와 보안의 핵심 원칙을 유지하면서 블록체인 애플리케이션의 대중화된 글로벌 사용을 가능하게 하는 것입니다. 대규모 채택은 기술적 실행뿐만 아니라 사용자 경험과 비용 효율성 측면에서도 확장성을 필요로 합니다.

일반적인 확장 기술은 아래에 나열되어 있습니다:

Monolithic Layer-1 Blockchains

단일 레이어-1 블록체인은 하드웨어 발전을 위한 경쟁에 참여하고 있습니다. 단일 레이어 내에서 트랜잭션 처리의 모든 측면을 처리하도록 설계된 이러한 블록체인은 처리량 개선에 효과적임을 보여주었습니다. 그러나 향후 2년 동안 예상되는 하드웨어 발전의 유한한 용량으로 인해 임계 제한에 직면할 것입니다. 블록체인 애플리케이션에 대한 수요가 증가함에 따라 이러한 내재된 한계는 더욱 두드러져 확장성을 제한하게 됩니다.

Layer-2 Technologies

레이어-2 솔루션은 일반적으로 ZK 롤업과 Optimistic 롤업의 두 가지 범주로 나뉩니다. 이러한 솔루션은 이더리움과 같은 기존 레이어-1 체인 위에 구축하여 블록체인 네트워크를 확장하는 것을 목표로 합니다. 이러한 기술은 특히 탈중앙화 금융(DeFi)에서 특정 사용 사례에 대한 더 높은 거래 처리량과 비용 절감을 용이하게 합니다. 그러나 상당한 상쇄 효과를 낳기도 합니다. 레이어-2 솔루션에 사용되는 중앙 집중식 시퀀서는 분산화에 걸림돌로 작용될 수 있으며, 메인넷에서 거래를 컨펌해야 하는 필요성으로 인해 비용과 용량 제한이 증가합니다. 게다가 거래 컨펌에 대한 지연 문제와 사기를 방지하지 못하는 경우가 많아 이러한 시스템의 무결성과 효율성이 저하됩니다. 사기 방지 및 분산 시퀀싱에 대한 초기 제품 사양이 레이어-2 생태계에서 실현되는 것을 아직 보지 못했습니다.

Modularity in Decoupling Consensus from Data Availability

합의 메커니즘을 데이터 가용성과 분리하는 모듈화는 확장성에 기여할 수 있는 가장 유망한 접근 방식입니다. 이러한 기능을 분리함으로써 블록체인은 더 큰 효율성과 유연성을 달성할 수 있습니다. 그러나 해당 접근 방식은 개발자와 최종 사용자 모두에게 복잡하고 난해한 경험을 증가시키며, 사용자 경험(UX) 문제와 유동성의 분산을 가져옵니다. 이러한 요인들은 모듈러 시스템의 원활한 채택과 통합을 저해할 수 있습니다.

Modularity with Independent Shards/ Blockchain Modules

또 다른 모듈화된 접근 방식은 풀링되거나 공유된 기본 보안을 공유하면서도 독립적으로 작동하는 수많은 샤드나 모듈을 생성하는 것입니다. 이 방법은 여러 독립적인 유닛에 작업 부하를 분산시킴으로써 확장성을 향상시킵니다. 그러나 이러한 샤드들의 보안과 상호 운용성을 보장하는 것은 어려울 수 있으며, 샤드 아키텍처를 관리하는 난이도는 광범위한 사용과 채택에 추가적인 장애물이 될 수 있습니다.

Layer 1 Parallelization

블록체인 네트워크의 병렬화를 통해 트랜잭션을 동시에 처리하여 사용 가능한 모든 네트워크 리소스를 활용할 수 있습니다. 여러 상태에서 트랜잭션을 동시에 처리함으로써 네트워크는 궁극적으로 노드가 합의에 도달하기 위해 합의한 일치된 상태에 도달합니다. 이는 트랜잭션이 차례로 처리되는 이전 블록체인 시스템에서 사용하는

순차적 실행 모델과 다릅니다. 이러한 기술은 유망하지만 출시가 느리고 아직 상당한 온체인 트랜잭션을 생성하지 못했습니다.

이러한 각 기술은 블록체인 확장성의 한계를 해결하려하지만, 고유한 약점도 있습니다. 따라서 최적의 솔루션은 기술과 아키텍처의 조합에 있다고 제안합니다. 스케일의 하이브리드 접근 방식은 보안이나 분산화를 손상시키지 않고 블록체인 기술의 대량 채택을 가능하게 하는 궁극적인 목표를 달성하도록 설계되었습니다.

"Built Different" – SKALE The Gas Free, Invisible Blockchain for Mass Adoption

SKALE은 6가지 핵심적인 검증된 혁신을 통해 대중적인 채택을 가능하게 하기 위해 설계된 레이어-1 블록체인 네트워크입니다. 궁극적으로 SKALE은 풀링된 보안 속성과 이더리움 블록체인으로부터의 보안 앵커를 갖춘 레이어-1 블록체인(샤드)의 네트워크입니다. SKALE은 사이드체인이 아니며, SKALE 체인들은 보안상의 결함이 잘 알려진 사이드체인이 아닙니다. SKALE의 아키텍처는 모듈들이 ZK 롤업 기능과 통합될 때 독립적인 레이어-1 체인이나 레이어-2 체인으로 작동할 수 있도록 합니다. 이 아키텍처는 네트워크가 존재한 첫 4년 동안 7억 건 이상의 온체인 트랜잭션을 4,500만 개 이상의 고유 활성 지갑에 대해 처리하면서 그 실효성이 입증되었습니다.

Gas-Free Transactions

SKALE은 다음과 같은 메커니즘을 통해 수수료(Gas) 없는 트랜잭션 모델을 도입합니다:

- **Unified Validation at Protocol Consensus Level:** 통합 검증은 모든 검증자가 팀으로서 함께 일한다는 것을 의미합니다. 각 블록으로 수수료를 놓고 경쟁하는 대신, 그들은 하나의 단위로 일하며 수수료와 인플레이션을 월 단위로 지불하여 동등하게 공유합니다. 그들은 팀으로서 생산하고 검증할 뿐만 아니라, 정직하고 성과가 좋은 검증자 노드만이 월 단위로 수익을 나누도록 팀으로서 서로를 모니터링합니다. 이를 통해 보다 효율적이고 간소화된 합의 프로세스가 가능합니다.
- **Dynamic Chain Fees:** SKALE은 기존 가스 수수료 대신 동적 가스 수수료와 관련된 UX 문제를 최종 사용자에게 부과하지 않고 건전한 네트워크 경제성을 유지하는 동적 체인 수수료를 사용합니다. 이러한 방식으로 최종 사용자가 아닌 개발자, 애플리케이션 및 기업이 인프라 구독료를 지불하는 Web2와 유사한 방법을 사용 할 수 있습니다.
- **sFUEL:** sFUEL은 가스와 유사하게 작동하지만 최종 사용자에게 경제적 비용을 발생시키지 않습니다. sFUEL의 희소성과 지능적인 분배는 잠재적 공격자가 네트워크를 공격하거나 마비시키기 전에 sFUEL이 빠르게 소진하게 됨으로 DDoS 및 스팸 공격을 방지합니다.

Pooled Security Shards

SKALE은 체인 수준에서의 모듈러성을 활용하여 풀링된 보안 샤드를 생성합니다:

- **Independent Shard Operation:** 각 샤드는 운영적인 관점에서 독립적으로 동작하지만, 단일 검증자 네트워크를 통해 보안을 공유합니다.
- **Shared Security without Shared Capacity:** 각 샤드는 본질적으로 자체적인 레이어-1 블록체인입니다. 한 샤드의 혼잡이나 사용량은 다른 샤드의 성능이나 가격에 전혀 영향을 미치지 않습니다. 샤드를 운영하는 노드는 네트워크 전체 노드 풀에서 동적으로 변하는 하위 집합으로 선택됩니다.
- **Elastic Supply**: 네트워크에 더 많은 용량이 필요할 때 샤드를 쉽게 생성할 수 있습니다.
- **Appchains or HubChains**: 샤드는 개별 앱체인으로 작동하거나 여러 애플리케이션에서 동시에 사용되는 공유 체인인 허브체인으로 작동할 수 있습니다.

Root Security Anchor

SKALE은 공유 보안의 핵심 원칙을 통해 이더리움 네트워크에 보안을 앵커링합니다:

- **Root Staked Token:** 루트 스테이킹 토큰은 앵커 체인(이더리움)에서 생성되어 스테이킹되며, 네트워크를 안전하게 보호합니다.
- **Shard Functionality Anchored into Ethereum Security:** 이는 샤드 보안과 담합을 방지하는 네트워크 운영이 SKALE 검증자 네트워크가 아닌 앵커 체인에서 관리되도록 보장합니다.

Optimized Ethereum Virtual Machine

SKALE은 하이퍼-고급 C++ 이더리움 가상 머신(EVM)을 실행합니다:

- **Enhanced Throughput:** 이는 EVM 환경 내에서 합의 과정을 훨씬 더 빠르게 처리할 수 있게 합니다. 많은 고속 합의 알고리즘이 표준 gETH EVM 포크에 의해 제한되지만, SKALE의 첨단 EVM은 이러한 제한을 우회합니다.

Shard/Chain Interoperability

SKALE 샤드들은 서로 간에 거의 즉각적으로 메시지를 주고받거나 브리지할 수 있으며, 수수료가 전혀 없습니다.

SKALE IMA 브리징은 세계 최초의 라이브 BLS 임계값 암호화 브리지로서, 체인 간의 거의 즉각적이고 무료인 브리징을 제공합니다. 이 다이내믹한 연결은 SKALE 네트워크를 통합하고, 가스가 필요 없는 브리징, 맞춤형 스마트 컨트랙트, 그리고 사용자 경험에 최적화된 브리지 UI를 통해 샤딩의 복잡성을 최종 사용자가 겪지 않도록 해결합니다. 이를 통해 사용자는 샤딩, 체인, 유동성, 토큰 매핑 등의 복잡성을 이해할 필요 없이 생태계 전반에서 편리하게 참여할 수 있습니다.

이러한 혁신을 통해, SKALE은 탈중앙화나 보안을 타협하지 않으면서 대중적인 채택을 지원하는 확장 가능하고 안전하며 사용자 친화적인 블록체인 플랫폼을 제공하는 것을 목표로 합니다.

SKALE Manager

SKALE 루트 보안 앵커(Root Security Anchor)는 Ethereum 메인넷에서 실행되는 스마트 계약으로 구성되며, 이를 통틀어 SKALE Manager라고 합니다. SKALE Manager는 검증자 등록, 체인 생성/파괴, 슈퍼 노드 선택, 슈퍼 노드 로테이션, 스테이킹, 바운티 지급, 인출, 슬래싱 등과 같은 SKALE 네트워크의 중요한 기능을 운영합니다.

Super Node Creation

시스템에 슈퍼 노드로 추가되기 위해서는, 예비 슈퍼 노드가 SKALE 데몬을 실행해야 합니다. 이 데몬은 해당 노드가 네트워크의 하드웨어 요구 사항[1]을 준수하는지 평가합니다. 예비 슈퍼 노드가 이 검증 단계를 통과하면, 데몬은 네트워크에 가입하기 위한 요청을 SKALE 매니저에게 제출할 수 있도록 허용합니다. 이 요청에는 필요한 네트워크 예치금과 데몬이 수집한 노드 메타데이터(예: IP 주소, 포트, 공개 키 등)가 포함됩니다. 요청이 이더리움에 커밋된 후, 예비 슈퍼 노드는 시스템에 '풀 노드' 또는 '프랙셔널 노드'[2]로 추가됩니다. 풀 노드는 모든 자원을 단일 SKALE 체인에 활용하며, 프랙셔널 노드는 여러 SKALE 체인에 참여합니다(멀티 테넌시).

[1] 노드 운영에 관심이 있는 당사자들의 진입 장벽을 낮추기 위해 이러한 하드웨어 요구 사항은 변경될 수 있습니다.
[2] 전체 노드와 부분 노드의 비율은 시장 수요에 따라 결정될 것이며, 실제 알고리즘은 추후에 지정될 예정입니다.

슈퍼 노드가 생성되면 네트워크의 대규모 피어 슈퍼 노드 그룹[3]이 무작위로 할당되며, 피어는 미리 정해진 주기(예: 5분)에 슈퍼 노드 다운타임과 지연 시간을 정기적으로 감사하고 이러한 일괄 지표를 슈퍼 노드의 현상금 보상을 결정하는 데 사용되는며, 네트워크 기간마다 한 번씩 SKALE 관리자에게 제출합니다.

Super Node Destruction

네트워크에서 퇴출할 때, 슈퍼 노드는 먼저 자신의 퇴출을 선언하고 확정 기간(finalization period)[4]을 기다려야 합니다. 이 확정 기간(예: 2일)이 지나면, 해당 노드는 비활성화되며 네트워크에서 초기 스테이킹한 자금을 인출할 수 있습니다.

사용자가 확정 기간을 기다리지 않고 네트워크에서 즉시 노드를 종료하는 경우, SLA 노드에 의해 비준수(죽은) 노드로 분류되고 슈퍼 노드에 대한 보상금은 지급되지 않습니다. 그 노드는 다음 체인에서 순환되도록 대기하게 됩니다.

SKALE Chain Creation

SKALE 체인이 생성될 때, 새로운 체인 소유자는 체인의 구성(config)을 선택하고 SKALE 체인의 생성을 요청합니다. 이 작업이 완료되면 체인 소유자는 현재 에포크(기간)가 끝나기 전에 SKALE 유로파 허브의 SKALE 체인 가격 결제원에 결제해야 합니다. 제때 결제하지 않은 체인은 삭제될 수 있습니다

자원 효율성을 최대화하기 위해, 체인 소유자들은 소형, 중형, 대형 중에서 자신의 SKALE 체인 크기를 선택할 수 있습니다. 최소한 16개의 노드가 무작위로 각 체인에 할당되어 검증을 수행하며, 각 노드는 체인 크기에 따라 지정된 자원을 활용합니다.

- 소형 체인은 노드 자원의 1/128을 사용합니다.
- 중형 체인은 노드 자원의 1/16을 사용합니다.
- 대형 체인은 노드의 모든 자원(단, SKALE 데몬을 실행하는 자원 제외)을 사용합니다.

> 현재 네트워크의 모든 자원은 동일한 가치가 있으며, 이러한 자원을 소비하는 비용은 체인의 크기에 따라 다릅니다. 네트워크가 성숙해짐에 따라, 네트워크 자원의 비용은 현재 네트워크 상태, 시스템 부하 및 기타 가변적인 요소를 고려하여 동적으로 계산될 수 있습니다. 또한, 체인 생성 프로세스는 사용자가 노드 수, 서명자 수, 합의 알고리즘, 기타 합의/체인 기반 변수 등을 지정하여 체인의 추가적인 측면을 사용자 정의할 수 있도록 허용합니다.

SKALE 관리자가 생성 요청을 받은 후 새 SKALE 체인이 생성되고 각 엔드포인트가 체인 소유자에게 돌아가게됩니다. 네트워크에 원하는 SKALE 체인 생성을 지원할 리소스가 부족하면 거래가 취소되고 체인 개설자에게 통지됩니다.

Unified Validation, Random Selection, and Frequent Node Rotation

SKALE 네트워크는 지분 증명(Proof of Stake) 운영 환경입니다. 지분 증명 네트워크에서 거래의 보안성과 유효성은 주로 검증자 노드의 성능, 아키텍처, 그리고 밸리데이터 노드의 동태에 의존합니다. 검증 레이어가 적절히 작동하도록 보장하려면 네트워크에 많은 수의 검증자 노드가 필요합니다. 네트워크 내 노드 수가 적으면 본질적으로 위험에 취약합니다.

---

[3] 피어 노드의 수는 24개를 목표로 하지만, 이는 변경될 수 있습니다.
[4] 노드의 피어들이 해제되고, SKALE 체인에 새로운 노드들이 임명되는 기간입니다.

안전하고 견고한 네트워크를 구축하고 운영하기 위해서는 인프라의 기반이 다음 두 가지에 의존합니다: a) 체인 검증자 세트의 무작위 선택, b) 체인 내외로 노드의 빈번한 회전. 무작위성과 회전이 없다면, 검증자 간의 뇌물 수수 또는 공모의 위험이 증가하여 네트워크 내 체인의 보안성과 무결성이 크게 감소합니다.

SKALE 네트워크의 검증자 노드들은 독립적인 방식으로 운영되고 조직됩니다. 어떤 체인에서 작업할지를 선택하는 중앙 노드 컨트롤러는 존재하지 않습니다. 대신 노드들은 커뮤니티로서 알고리즘에 협력하여 무작위성과 회전을 향상시켜 어떤 체인에서 작업할지를 결정합니다. 각 노드는 드론과 유사하게 동작합니다. 노드는 분산되었지만 조정되고 통합된 방식으로 네트워크의 다른 노드들과 병렬로 SKALE 매니저 메인넷과 단독으로 통신합니다.

지분 증명 네트워크의 마지막 요구 사항은 페널티와 보상을 모두 아우르는 적절한 인센티브 구조입니다. 전자와 관련하여, 모든 검증자 노드는 네트워크에 상당한 가치를 스테이킹해야 합니다. 스테이킹은 올바른 행동을 보장하는데, 만약 검증자가 공모하거나 비잔틴 행동을 하는 것으로 판명되면, 그 검증자는 스테이킹한 금액을 몰수당하고 네트워크에서 제거되게 됩니다.

무작위 선택과 빈번한 노드 회전을 사용하는 이러한 유형의 풀링된 검증 모델에서, 악의적인 행위자가 SKALE 체인의 검증자들을 강압하거나 뇌물을 주기 위해서는 전체 네트워크의 2/3를 효과적으로 매수해야 합니다. 현재 그리고 네트워크가 성장함에 따라, 현재 네트워크의 규모를 고려하면 이를 달성하기는 매우 어려울 것입니다. SKALE의 네트워크 디자인은 이러한 핵심 원칙에 기반을 두고 있으며, 네트워크 내 각 체인의 거래 무결성을 유지하면서 공격을 방지하고 제거하는 것과 직접적으로 일치합니다.

SKALE Chain Destruction

SKALE 체인의 파기는 체인 소유자의 네트워크 리소스 비용 지불이 3개월 이상 연체되었거나 체인 소유자가 자신의 SKALE 체인을 삭제 대상으로 지정한 경우에 발생합니다. 체인 비용 지불이 SKALE 체인 가격 정책에 따라 지불되지 않으면, 체인 소유자는 예정된 체인 삭제에 대해 통보를 받고 네트워크에 대한 부채를 상환하고 추가 시간을 선결제할 기회를 받게 됩니다.

파기 프로세스에서는 이더리움에서 기원한 모든 암호화 자산을 메인넷에서 소유자에게 전송하고, SKALE 체인의 모든 노드를 제거하며, 스토리지와 메모리를 하드 리셋하고, SKALE 매니저에서 SKALE 체인이 제거된 후 체인 파기를 요청한 제출자에게 보상을 제공합니다.

> *Note:* 제출자가 체인을 파괴하여 얻는 보상은 거래 비용보다 약간 크며 네트워크 리소스에 인센티브를 제공하는 쓰레기 수거 메커니즘 역할을 합니다.

Validator Rewards and Bounty Issuance

제안된 내용: SKALE 체인의 설립은 네트워크의 핵심 대표단과 독립적입니다. 이를 통해 검증자의 수익성이 향상되는 동시에 검증자 인센티브를 위해 대표단이 감소하지 않도록 보장할 수 있습니다. 네트워크에는 모든 체인이 그들의 체인을 활성화하기 위해 따라야 하는 임대 시스템이 있습니다. 이러한 시스템의 이점은 검증자가 매달 체인 수수료를 지급받고 인센티브 패키지의 일부로 청구할 수 있다는 것입니다.

또한 매월 인플레이션(발행) 이벤트가 발생하여, 이더리움 메인넷의 스마트 컨트랙트를 통해 새로운 SKALE 토큰이 생성됩니다. 생성된 토큰은 검증자들에게 지급되는 바운티 풀에 추가됩니다. 예를 들어, 네트워크에 천 개의 검증자 노드가 있고 모두 성능이 우수하다면, 그들은 체인 토큰 스테이킹의 일부와 인플레이션 금액으로 구성된 바운티 풀의 월간 수익을 공유하게 됩니다.

SKALE Chain Operations

SKALE Consensus

SKALE 네트워크는 합의 모델로 비동기 2진 비잔틴 합의(ABBA) 프로토콜의 변형을 사용합니다. ABBA 프로토콜의 근본적인 이점은 각 지연되거나 다운된 노드를 느린 링크로 간주하여 노드 다운타임의 경우에도 강건성을 발휘하도록 설계되었다는 것입니다. ABBA 기반의 합의 계층은 각 노드 내에 컨테이너로 포함된 EVM(이더리움 가상 머신)의 기본 합의 모델을 대체합니다. 이 EVM이 각 노드 내에서 인스턴스화되어 각 SKALE 체인을 관리하고 유지합니다. SKALE 체인은 이 비동기적이고 리더가 없으며 수학적으로 안전한 ABBA 프로토콜의 변형을 사용하여 블록 생성과 커밋에 참여하는 노드 집합을 통해 검증됩니다.

SKALE 네트워크의 합의 알고리즘은 합의 기반 체인 검증과 생성에 대한 수학적으로 증명 가능한 접근법을 제시하기 위해 학술적 연구에 크게 의존합니다. 산업이 이 정도로 발전한 현재에도, 블록체인 세계에서는 결과를 우선시 하는 시스템과 학술 연구 사이에 분열이 있습니다. 전자에는 비트코인과 이더리움 1.0 같은 작업 증명 네트워크와 코스모스와 텐더민트 같은 지분 증명 시스템이 있습니다. 학술 분야에서는 ABBA 기반 접근법과 수학적으로 증명 가능한 여러 알고리즘이 존재합니다.

SKALE 네트워크의 설계는 실제 트레일과 수학적 연구 사이의 이러한 격차를 해소합니다. SKALE은 상업적으로 이용 가능한 대규모 생산 시스템에서 2진 합의를 활용한 최초의 네트워크입니다. 이는 다른 생산 알고리즘에 결함이 있음을 암시하거나 추론하기 위한 것이 아니라 수학적으로 증명 가능한 알고리즘을 사용하기 위한 엄격함에 의해 SKALE 컨센서스의 설계가 유도된다는 점을 강조하기 위한 것입니다.

전체 노드 검증자 집합의 2/3 이상이 온라인 상태인 한, 그들은 계속해서 체인에 새로운 블록을 생성하고 커밋할 것입니다. 이 프로토콜은 아래의 다이어그램으로 설명되며, 다음 섹션에서 자세히 다뤄집니다.

Threshold Signatures

우리의 프로토콜은 슈퍼다수결 투표를 위해 임계 서명을 사용합니다. SKALE 체인이 생성될 때, Boneh-Lynn-Shacham(BLS) 개인 키 공유 PKS[I]가 Joint-Feldman 분산 키 생성(DKG)을 사용하여 생성되어 각 노드에 발급됩니다. 각 PKS[I]에 대해, 검증 가능한 공개 키 PK[I]가 존재하며, 이는 서명 검증을 위해 SKALE 매니저에 저장되어 공개적으로 이용 가능합니다. BLS 임계 서명은 볼디레바(Boldyreva)의 설명에 따라 구현되며, 이더리움 콘스탄티노플 릴리스에서 구현된 타원 곡선(altBN256)과 그룹 페어링(Optimal-Ate)을 사용합니다.

Threshold Encrypted Messaging

Network Security Assumptions

이 프로토콜은 네트워크가 비동기적이며 최종적인 전달이 보장된다고 가정합니다. 이는 모든 노드가 신뢰할 수 있는 통신 링크로 연결되어 있다고 가정하며, 링크는 임의로 느릴 수 있지만 결국에는 메시지를 전달한다는 의미입니다.

이러한 비동기 모델은 비트코인과 이더리움 블록체인과 유사하며, 일시적인 네트워크 분할이 일반적이지만 결국 해결되는 현대 인터넷의 상태를 반영합니다. 최종적인 전달 보장은 실제로 송신 노드가 지수적 백오프를 사용하여 수신 노드로 메시지를 전송하는 여러 시도를 통해 달성되며, 성공할 때까지 계속 시도합니다.

Pending Transactions Queue

각 노드는 대기 트랜잭션 대기열[5]를 유지합니다. 이 대기열에 트랜잭션을 처음으로 받은 노드는 해당 트랜잭션을 각 피어별로 전용 출력 메시지 대기열을 통해 피어들에게 전파하려고 시도합니다. 특정 피어에게 메시지를 전달하도록 스케줄링하려면, 해당하는 출력 대기열에 메시지를 배치합니다. 이러한 출력 대기열 각각은 별도의 스레드에 의해 관리되며, 이를 통해 메시지가 병렬로 전달되어 특정 피어가 메시지 수신을 거부하더라도 다른 피어들의 메시지 수신에 영향을 미치지 않습니다.

Block Proposal

이전 합의 라운드가 완료되면, 각 노드의 TIP_ID는 1 증가하며 즉시 블록 제안을 생성하게 됩니다.

블록 제안을 생성하기 위해, 노드는 다음을 수행합니다:

1. 자신의 대기 트랜잭션 큐를 확인합니다.
2. 대기 큐의 트랜잭션 총 크기가 MAX_BLOCK_SIZE보다 작거나 같은 경우, 노드는 큐에서 모든 트랜잭션을 가져와 블록 제안을 채웁니다.
3. 대기 큐의 트랜잭션 총 크기가 MAX_BLOCK_SIZE를 초과하는 경우, 노드는 가장 오래된 것부터 가장 최근에 수신된 순서대로 대기 큐에서 트랜잭션을 가져와 MAX_BLOCK_SIZE만큼 블록 제안을 채웁니다.
4. 노드는 트랜잭션을 SHA-256 머클 루트의 작은 값에서 큰 값 순으로 정렬하여 블록 제안을 구성합니다.
5. 대기 큐가 비어 있는 경우, 노드는 BEACON_TIME 동안 기다린 후, 큐가 여전히 비어 있으면 트랜잭션이 없는 빈 블록 제안을 만듭니다.

   Note: 노드는 제안 시점에 보류 대기열에 있는 거래 내역을 임의로 제거하지 않습니다. 그 이유는 제안 시점에 제안이 수락될 것이라는 보장이 없기 때문입니다.

Data Availability

노드가 블록 제안을 생성하면, 아래에 설명된 데이터 가용성 프로토콜을 사용하여 다른 노드에 이를 전달합니다. 데이터 가용성 프로토콜은 메시지가 슈퍼다수결의 노드에게 전달되는 것을 보장합니다.

다음은 5단계의 프로토콜입니다:

1. 송신 노드 A는 블록 제안과 그 제안 P를 구성하는 트랜잭션의 해시를 모든 피어들에게 보냅니다.
2. 수신한 각 피어는 해시를 자신의 대기 큐의 트랜잭션과 매칭하여 P를 재구성합니다. 대기 큐에서 찾을 수 없는 트랜잭션에 대해서는 피어가 송신 노드 A에게 요청을 보냅니다. 그러면 A는 이러한 트랜잭션의 본문을 수신 노드에 보내며, 이를 통해 피어는 블록 제안을 재구성하고 제안을 자신의 제안 저장소 데이터베이스 PD에 추가합니다.
3. 피어는 P에 대한 임계 서명 공유를 포함한 영수증을 A에게 다시 보냅니다.
4. 노드 A는 슈퍼다수결(자신을 포함하여 2/3 이상)의 노드로부터 서명 공유를 수집할 때까지 기다립니다. 그런 다음 A는 슈퍼다수결 서명 S를 생성합니다. 이 서명은 슈퍼다수결의 노드들이 P를 보유하고 있다는 영수증 역할을 합니다.
5. A는 이 슈퍼다수결 서명 S를 네트워크의 다른 모든 노드에 브로드캐스트합니다.

   Note: 각 노드는 *BLS* 개인 키 공유 *PKS[I]*를 보유하고 있습니다. 키 공유의 초기 생성은 *Joint-Feldman* 분산 키 생성*(DKG)* 알고리즘을 사용하여 수행되며, 이는 *SKALE* 체인의 생성 시와 노드가 셔플될 때 발생합니다.

[5] 각 사용자의 트랜잭션들은 바이트 시퀀스로 표현되는 이더리움 호환 트랜잭션으로 간주됩니다.

추가 합의 단계에서는 제안 P에 투표하는 모든 노드에서 데이터 가용성 영수증이 필요하며, 이를 위해 투표에 초대다수 서명 S를 포함해야 합니다. 정직한 노드는 초대다수 서명 S를 포함하지 않는 모든 투표를 무시합니다. 이 프로토콜은 데이터 가용성을 보장하므로 합의에 도달한 모든 제안 P는 모든 정직한 노드에서 사용할 수 있습니다.

Pluggable Binary Byzantine Agreement

아래에 설명된 합의는 비동기 2진 비잔틴 합의(ABBA) 프로토콜을 사용합니다. SKALE은 현재 Mostefaoui 등으로부터 파생된 ABBA의 변형을 사용하고 있습니다. 다음의 속성을 만족하는한, 다른 ABBA 프로토콜 P를 사용할 수 있습니다:

- Network model: P는 위에서 설명한 비동기 네트워크 메시징 모델을 가정합니다.
- Byzantine nodes: P는 비잔틴 노드가 3분의 1 미만이라고 가정합니다.
- Initial vote: P는 각 노드가 초기 투표로 예(1) 또는 아니오(0)를 한다고 가정합니다.
- Consensus vote: P는 예 또는 아니오 중 하나의 합의 투표로 종료되며, 합의 투표가 Yes인 경우 적어도 하나의 정직한 노드가 예로 투표했음을 보장합니다.

  Note: ABBA 프로토콜은 일반적으로 그 작동의 부산물로서 난수 *COMMON_COIN*을 출력합니다. 우리는 이 *COMMON_COIN*을 난수 소스로 사용합니다.

Consensus Round

제안 단계가 완료되자마자, 제안 P에 대해 슈퍼다수결 서명 S를 받은 각 노드 A는 합의 라운드 R에서 비동기 비잔틴 2진 합의(ABBA)에 투표합니다. 프로토콜은 다음과 같습니다:

1. 각 R에 대해, 노드들은 N개의 ABBA 인스턴스를 실행합니다.
2. 각 ABBA[i]는 노드 i의 블록 제안에 대한 투표에 해당합니다.
3. 각 ABBA[i]는 예 또는 아니오의 합의 투표로 완료됩니다.
4. 모든 ABBA[i]가 완료되면, 각 제안에 대한 예 또는 아니오를 포함하는 투표 벡터 v[i]가 생성됩니다.
5. 예 투표가 하나만 있는 경우, 해당 블록 제안 P가 SKALE 체인에 커밋됩니다.
6. 예 투표가 여러 개인 경우, 의사난수 R을 사용하여 예 투표된 제안들 중에서 P를 무작위로 선택합니다. 승리한 제안은 R을 N_WIN으로 나눈 나머지에 해당하며, 여기서 N_WIN은 예 투표된 제안의 총 수입니다.
7. 난수 R은 모든 ABBA COMMON_COIN의 합입니다.
8. 모든 투표가 아니오인 드문 경우에는 빈 블록이 블록체인에 커밋됩니다. 모든 투표가 아니오일 확률은 매우 작으며, N이 증가함에 따라 감소합니다.

Finalizing Winning Block Proposal

어떤 노드 A에서 승리한 블록 제안 P로 합의가 완료되면, 노드는 제안을 마무리하고 체인에 커밋하기 위해 다음 알고리즘을 실행합니다:

1. 노드 A는 승리한 제안 P를 받았는지 확인합니다.
2. 노드 A가 제안을 받지 못한 경우, 피어 노드로부터 이를 요청하여 다운로드합니다.
3. 노드 A는 P에 대한 서명 공유 S를 서명하고, 이를 다른 모든 노드에 보냅니다.
4. 노드 A는 자신을 포함한 슈퍼다수결의 노드로부터 서명 공유를 받을 때까지 기다립니다.
5. 노드 A가 슈퍼다수결의 서명 공유를 받으면, 이를 임계 서명으로 결합합니다.
6. 노드 A는 P를 임계 서명 S와 함께 블록체인에 커밋합니다.

SKALE 체인에 커밋된 블록은 블록 헤더와 블록 본문을 포함합니다. 블록 본문은 블록 내 모든 트랜잭션의 연결된 트랜잭션 배열이며, 블록 헤더는 다음을 포함하는 JSON 객체입니다::

Name	Data Type	Description
BLOCK_ID	integer	현재 블록의 ID로, 0부터 시작하여 1씩 증가합니다.
BLOCK_PROPOSER	integer	블록을 제안한 노드의 ID입니다.
PREVIOUS_BLOCK_HASH	string	이전 블록의 SHA-256 머클 루트입니다.
CURRENT_BLOCK_HASH	string	현재 블록의 SHA-256 머클 루트입니다.
TRANSACTION_COUNT	integer	현재 블록의 트랜잭션 개수입니다.
TRANSACTION_SIZES	integer[]	현재 블록의 트랜잭션 크기 배열입니다.
CURRENT_BLOCK_PROPOSER_SIG	string	현재 블록 제안자의 ECDSA 서명입니다.
CURRENT_BLOCK_TSIG	integer	현재 블록의 BLS 슈퍼다수결 임계 서명입니다.

Table 1. SKALE 블록 헤더 포맷

SKALE Super Nodes and Nodes

각 SKALE 체인은 SKALE 데몬을 실행하고 SKALE 합의를 수행하는 무작위로 지정된 노드들의 집합으로 구성됩니다. 다른 프로토콜과는 달리, SKALE 노드들은 네트워크에 참여하는 노드들 사이에 일대일 매핑으로 제한되지 않습니다. 이는 SKALE 네트워크의 각 노드에 배포된 컨테이너화된 노드 아키텍처를 통해 가능해지며, 각 노드는 여러 SKALE 체인을 동시에 실행할 수 있습니다.

SKALE 노드 내의 노드는 슈퍼 노드 또는 노드라고 불립니다. 각 노드는 독립적인 SKALE 체인에 참여합니다. 아래는 SKALE 노드에서 실행되는 컨테이너의 다이어그램입니다.

이 컨테이너화된 아키텍처는 탄력성, 구성 가능성 및 모듈성을 제공하는 중앙 집중식 시스템과 동등한 엔터프라이즈급 성능과 다양한 선택의 폭을 분산형 애플리케이션 개발자에게 제공하는 수단으로 선택되었습니다. 컨테이너는 도커화된 Linux OS와 함께 제공되는 5가지 주요 구성 요소로 나뉩니다. 각 노드는 OS에 독립적인 방식으로 호스팅될 수 있습니다. 각 컨테이너는 다음 서비스 중 하나에 캡슐화됩니다.

SKALE Admin Service

SKALE 관리자 서비스는 이더리움 메인넷에 위치한 SKALE 매니저와 노드 간의 사용자 인터페이스 역할을 합니다. 이 인터페이스를 통해 노드는 자신이 참여하고 있는 SKALE 체인을 확인하고, SKALE 토큰의 입금, 출금, 스테이킹 및 클레임[6]을 수행할 수 있습니다. 슈퍼 노드 내의 노드들은 SKALE 체인에 무작위로 할당되기 때문에, 네트워크 내에서 SKALE 체인에 참여하거나 탈퇴할 수 있는 인터페이스는 제공되지 않습니다.

Node Monitoring Service

NMS(Node Monitoring Service)는 각 SKALE 노드에서 실행되며, 해당 노드의 피어 노드들의 성능 추적을 용이하게 합니다. 성능 추적은 각 피어 노드에 핑을 보내고 업타임과 지연 시간을 측정하고, 이러한 측정치를

[6] 청구 가능한 SKALE 토큰은 네트워크 기간 동안의 평균 가동 시간/지연 시간을 기준으로 노드에 정기적으로 발행되는 토큰입니다.

로컬 데이터베이스에 기록합니다. 각 SKALE 체인 에포크가 끝날 때마다 이러한 측정 항목의 평균이 산출되어 SKALE 관리자에게 제출되고, SKALE 관리자는 이를 사용하여 각 노드에 대한 배당금을 결정합니다.

Node Orchestration Service

NOS(Node Orchestration Service)는 SKALE 데몬(skaled), SKALE 체인 동기화를 위한 캐치업 에이전트, 그리고 인터체인 메시징을 위한 전송 에이전트로 구성된 동적으로 생성된 노드 이미지를 사용하여 노드의 계산 및 스토리지 자원을 활용하여 노드를 인스턴스화합니다. 이 서비스는 실패한 노드의 재실행뿐만 아니라 해제된 노드에 대한 자원의 할당 해제도 수행합니다.

SKALE Chain Pricing

SKALE은 검증자 인센티브가 지속 가능하고 매력적이도록 보장하는 동시에, 거래당 지불 블록체인과 비교하여 리소스 가격의 균형을 맞추기 위해 지속 가능한 분산형 경제 모델을 갖추고 있습니다.

체인 가격 책정은 네트워크 자원을 안정적이고 일관된 지불로 임대할 수 있는 DSaaS(탈중앙화 서비스형 소프트웨어) 프로그램과 동등합니다.

체인 가격 책정은 다음과 같이 책정됩니다:

- - SKALE 체인은 네트워크 활용도에 따라 결정되는 고정 임대료가 적용됩니다. 네트워크 활용도는 SKALE 체인의 수 / (네트워크 내 총 노드 수 / 2)로 계산됩니다.
- - SKALE 체인 수수료와 분배 비율은 백서의 거버넌스 섹션에 설명된 대로 온체인 투표를 통해 수정될 수 있습니다.
- - SKALE 체인은 현재 가격으로 최대 24개월까지 선불로 결제할 수 있으며, 이는 체인 소유자를 단기 가격 변동으로부터 보호합니다.
- - SKALE 체인은 최대 3개월까지 부채 상태로 운영될 수 있으며, 그 이후에는 체인과 모든 관련 상태 및 정보가 네트워크에서 제거될 수 있습니다.

검증자들은 SKALE 체인 가격 책정을 통해 다음과 같은 혜택을 받습니다:

- 지불 기한 후 각 기간마다 월 임대료를 인출할 수 있습니다.
- SKALE Europa Hub에서 직접 SKL 토큰을 인출할 수 있어, 인출 시, 인출 가능한 수익을 줄일 수 있는 가스 수수료를 절약할 수 있습니다.

SKALE 체인 가격 책정은 모든 구성원이 운영 비용과 자원 사용을 공정하게 균형 잡을 수 있도록 하는 탈중앙화 네트워크를 위한 정교한 경제 모델을 가능하게 합니다. SKALE 체인 가격 책정에 대해 자세히 알아보려면 SKALE 포럼 게시물을 방문하세요.

Attacks and Faults

네트워크 다운타임을 고려하기 위해 SKALE은 노드와 체인 수준 모두에서 장애 복구를 수행하기 위한 일련의 비상 전략을 적용했습니다. 이러한 전략은 다운된 노드에 대한 복구를 수행하는 자동화된 에이전트부터 네트워크의 모든 SKALE 체인 운영자가 사용할 수 있는 보안 사고 대응 팀에 이르기까지 다양합니다.

Reboots / Crashes

재부팅 중에는 재부팅되는 노드는 일시적으로 사용 불가능 합니다. 피어 노드들에게는 이것이 일시적으로 느린 네트워크 링크처럼 보일 것입니다. 재부팅 후에는 해당 노드로 향하는 메시지가 전달되며, 이 프로토콜은 합의 작업을 방해하지 않고 재부팅이 이루어질 수 있도록 합니다.

하드웨어 장애나 소프트웨어 버그로 인해 노드가 온라인 상태가 되지 못해 합의 상태를 잃는 하드 크래시의 경우, 피어 노드들은 출력 메시지 대기열이 오버플로되어 될 때까지 메시지를 보내려고 계속 시도하여, 오래된 메시지가 삭제됩니다. 이러한 영향을 완화하기 위해 1시간 이상 된 메시지는 메시지 대기열에서 삭제되도록 합니다.

노드가 하드 크래시를 겪는 다면, 이 노드는 각 합의 라운드에서 비잔틴 노드로 간주되며, 동시에 전체 노드의 ⅓ 미만이 하드 크래시를 겪을 수 있도록 허용됩니다. 전체 노드의 ⅓ 이상이 하드 크래시를 겪는 경우, 합의가 중단되어 블록체인이 활성성을 잃을 수 있습니다.

이러한 치명적인 실패는 일정 기간 동안 새로운 블록 커밋이 없는 것을 통해 감지됩니다. 이 시점에서 이더리움 메인 체인을 활용한 실패 복구 프로토콜이 실행됩니다. 노드들은 합의 작업을 중지하고, 블록체인을 동기화하며, 합의를 재시작할 시간에 대해 결정합니다.

Catchup Agent

각 노드에서 실행되는 별도의 캐치업 에이전트는 해당 노드의 블록체인과 블록 제안 데이터베이스가 네트워크와 동기화되도록 보장하는 역할을 합니다. 캐치업 엔진은 다른 노드들과 지속적으로 무작위로 동기화 연결을 수행하며, 피어보다 작은 TIP_ID를 가진 노드는 누락된 블록을 다운로드하고, 수신한 블록의 슈퍼다수결 임계 서명을 검증한 뒤 자신의 체인에 커밋합니다.

노드가 심각한 장애로부터 온라인 상태로 복귀하면, 즉시 이 캐치업 절차를 시작하면서 이와 동시에 새로운 블록에 대한 합의에 참여합니다. 이때 블록 제안을 수락하고 합의 메커니즘에 따라 투표하게 되지만, 자체 블록 제안은 발행하지 않습니다. 그 이유는 각 블록 제안에 이전 블록의 해시가 필요한데, 노드는 캐치업 절차를 완료한 후에야 특정 블록 ID에 대한 자체 블록 제안을 발행할 수 있기 때문입니다.

각 노드에서 이러한 에이전트가 실행되므로, 심각한 장애를 경험한 노드도 체인을 재동기화한 후 블록 제안에 쉽게 다시 참여할 수 있습니다.

Security Incident Response

모든 탈중앙화 시스템에서 보안은 가장 중요한 요구 사항입니다. 그러나 암호학과 컴퓨터 과학의 발전에도 불구하고, 대부분의 보안 전문가들은 완벽한 보안은 달성 불가능하다고 동의합니다. 이를 감안할 때, 설계자들은 시스템을 무너뜨리는데 필요한 자원과 비용의 장벽을 최대한 높이는데 집중해야 합니다.

SKALE 아키텍처는 SKALE 체인을 기반으로 하기 때문에 SKALE에 보안이 훼손된다면 또 다른 SKALE 체인의 보안 침해가 수반될 수 있습니다. 예를 들어, Linux 커널의 버그로 인해 상당수의 노드가 컴퓨터 바이러스의 영향을 받을 수 있습니다. 이러한 경우 기본 절차는 다음과 같습니다:

1. 보안 침해가 의심되는 SKALE 체인 소유자는 이더리움의 SKALE 매니저 컨트랙트에 체인을 일시적으로 중지하도록 요청합니다.
2. SKALE 매니저는 해당 SKALE 체인이 중지된 것으로 표시합니다.
3. 손상되지 않은 노드들은 SKALE 매니저로부터 운영을 중지하라는 알림을 받습니다.
4. SKALE 체인의 클라이언트들은 중지 사실을 통보받고, 해당 SKALE 체인에 대한 요청이 거부됩니다.

SKALE Hubs

SKALE 허브는 여러 애플리케이션들이 함께 공유하는 SKALE 체인입니다.

SKALE 네트워크 허브는 다른 SKALE 체인에 핵심 서비스를 제공하는 동시에 유사한 유형의 애플리케이션을 위한 공유 컴퓨팅 자원으로도 작동합니다. 예를 들어, 이 글을 쓰는 시점에 네뷸라 허브(Nebula Hub)는 50개

이상의 게임의 기반이 되고 있습니다. 허브는 주요 네트워크 자원을 확장 가능한 방식으로 배포하여 많은 체인이 해당 자원을 활용할 수 있게 하며, 각 체인마다 배포할 필요가 없습니다.

모든 체인에 거래소나 마켓플레이스를 두는 대신 (이는 유동성이 분산되는 결과를 초래할 수 있음) 유동성 풀, 법정 화폐 온램프와 오프램프, 마켓플레이스 등이 SKALE 허브에서 제공되어 dApp과 다른 SKALE 체인이 이러한 핵심 자원을 활용할 수 있게 합니다.

SKALE 허브는 또한 dApp이 직접 구축할 수 있는 공유 컴퓨팅 환경을 제공합니다. SKALE 네트워크는 지속적으로 새로운 체인을 추가할 수 있지만, 대부분의 애플리케이션은 기존 SKALE 체인이 제공하는 자원의 일부분만 필요로 합니다. 이러한 이유로, 이러한 dApp은 SKALE 허브 중 하나에 직접 구축하는 것을 선택합니다.

멀티체인 아키텍처는 네트워크 용량과 확장성을 높이는 강력한 솔루션이지만, 체인 간 상호 작용은 지연된 응답과 높은 수수료 등 상당한 마찰을 일으킬 수 있습니다. 이러한 체인 간 마찰은 SKALE 네트워크의 모듈식 설계, 높은 수준의 암호화 및 합의 알고리즘의 사용, 그리고 가스비가 없는 설계를 통해 해결됩니다.

Extensions

이 네트워크 아키텍처와 프로토콜을 사용하면 여러 확장을 쉽게 추가하여 네트워크의 더 큰 기능과 확장성을 허용할 수 있습니다. 처음 구축된 두 가지는 각 노드 내의 향상된 파일 저장소와 SKALE 체인 간의 메시지를 중계하고 실행하기 위한 메커니즘을 포함합니다.

SKALE Block Storage and SKALE FileStorage

잠재적인 사용 사례를 확장하기 위해 SKALE은 기존 EVM을 수정하여 훨씬 더 큰 파일 저장 기능을 허용했습니다. 이러한 변화를 가능하게 한 요소로는 블록 크기의 증가(각 블록에 더 많은 데이터를 포함할 수 있도록)와 사전 컴파일된 파일스토리지 스마트 컨트랙트를 통해 각 노드의 파일 시스템에 직접 접근할 수 있게 한 것이 포함됩니다.

네트워크 사용자들은 파일을 1MB의 "청크"로 분할하여 파일스토리지 스마트 컨트랙트에 제출할 수 있으며, 이는 각 노드의 파일 시스템에 연속적으로 저장됩니다. 네트워크의 파일을 맞춤형 방식으로 삭제하여 추가 스토리지 기능에서 상태 과부하로 인해 네트워크가 리소스를 재할당할 수 있도록 할 수도 있습니다.

SKALE 네트워크는 또한 SKALE 파일스토리지 네트워크 서비스를 통해 안전한 노드 기반 파일 저장을 제공합니다. 버전 3.1 이후의 각 SKALE 체인은 체인 소유자의 필요에 따라 구성 가능한 저장 용량을 갖게 됩니다.

개발자들은 SKALE 체인에서 그들의 스마트 컨트랙트 내에서 이 스토리지를 활용할 수 있으며, 이 스토리지에 대한 모든 쓰기 작업은 SKALE 합의 모델을 통해 관리됩니다. 체인 소유자는 원한다면 이 데이터를 별도의 독립적인 탈중앙화 파일 스토리지로 자체적으로 유지할 수 있습니다. 이러한 원스톱 디지털 스토리지 기능은 타사 오프체인 솔루션을 혼합할 때 발생하는 마찰을 제거합니다. 이 스토리지가 사용될 수 있는 예로는 NFT 기반 이미지의 안전한 저장이 있는데, 이는 NFT 기반 애플리케이션이 겪는 가장 큰 과제 중 하나 입니다.

SKALE Interchain Messaging Agent (IMA Bridge)

SKALE 체인과 이더리움 메인넷 간, 그리고 SKALE 체인과 SKALE 허브 간의 메시징은 SKALE 인터체인 에이전트(또는 SKALE IMA 브리지)를 통해 처리됩니다. IMA 브리지 설계의 이점은 높은 보안성, 빠른 전송 속도, 자산 보관 유지, 비용 효율성, 그리고 수준 높은 맞춤화가 포함됩니다. IMA 브리지는 BLS 암호화, 이더리움 메인넷 스마트 컨트랙트, 지분 증명 합의 알고리즘, 그리고 그 외의 독특한 설계 요소를 활용하여 이러한 이점을 실현합니다. 이 브리지는 개발자와 사용자가 이더리움 메인넷과 모든 SKALE 체인 간, 또는 SKALE 체인과 허브

간에 디지털 자산을 안전하고 경제적으로 전송할 수 있도록 합니다. 이러한 디지털 자산에는 ETH, ERC-20, ERC-721, ERC-1155 토큰 뿐만 아니라 일반 메시징 데이터도 포함됩니다.

네트워크 브리지는 연결된 블록체인 솔루션 간의 필수 구성 요소로서, 네트워크 간 및 네트워크 내의 다른 체인 간에 디지털 자산의 안전한 전송을 가능하게 합니다. 이러한 브리지는 이러한 자산의 전송 에이전트 역할을 합니다. 그러나 실제 자산을 전송하는 대신, 이들은 잠금 박스와 프록시 제공자의 조합으로서 디지털 등가물의 검증된 전송을 제공합니다.

기본적으로 브리지는 사용자가 한 체인에서 다른 체인으로 임의의 메시지를 보내도록하여, 그 메시지의 내용이 두 번째 체인에서 기능적으로 작동하도록하지만, 이중 지출이나 두 체인에서의 별도의 동시 활동의 위험 요소는 발생하지 않게 합니다. 이같은 메시지 전달은 한 체인에서 메시지의 내용(예, 토큰)을 잠그거나 잠금 해제하고, 다른 체인에서 합성 자산을 제공함을 의미합니다.

SKALE IMA 브리지는 간단하고 투명하며 안전한 방식으로 작동합니다. 기본적인 거래 흐름은 다음과 같습니다.

- 토큰 보유자는 이더리움에서 SKALE 체인으로(또는 SKALE 체인에서 이더리움으로) 토큰 및 기타 디지털 자산을 전송하려는 의도를 보냅니다.
- 해당 SKALE 체인의 충분한 수의 검증인이 이 전송 신호를 온체인에서 확인하고, 선의이며 진실하다고 검증되면, 이더리움 메인넷(또는 반대로는 SKALE 네트워크)에서 거래가 시작됩니다. 자산은 보안이 유지되고, 메인넷의 SKALE 컨트랙트를 통해 이더리움 메인넷의 보관함에 보관됩니다.
- 이더리움 메인넷에서 정해진 블록 수(10개)가 지나면(거래가 완전히 인식되고 유효함을 보장하기 위해), 전송 요청이 SKALE IMA 에이전트라는 프록시로 보내집니다. 이 에이전트는 해당 SKALE 체인 내, 전송을 처리하기 위해 SKALE 노드 네트워크 내에서 작동하는 SKALE TokenManager를 호출합니다. 며칠이나 몇 시간이 아닌, 이 전송은 메인넷 블록 생성에 의해서만 버퍼링되어 빠르게 이루어집니다.

SKALE 체인에서 이더리움으로 가는 반대 방향의 전송도 유사한 방식으로 진행되지만, SKALE에서 이더리움 메인넷으로 나가는 경우에는 토큰이 SKALE 체인에서 소각되고 메인넷에서 잠금 해제됩니다. SKALE 체인에서 발행되고 이더리움으로 전송되는 토큰의 경우에는, 발행된 토큰의 소유권을 유지하는 방식으로 이더리움에서 발행 작업이 수행됩니다.

공개 키의 사용은 독립적인 SKALE 체인들이 다른 SKALE 체인에서 블록이 서명되고 커밋되었음을 검증할 수 있어, 스마트 컨트랙트의 실행뿐만 아니라 SKALE 체인 간의 디지털 자산 전송을 가능하게 합니다. 이 메커니즘은 이더리움 메인넷, 각 SKALE 체인에 위치한 일련의 스마트 컨트랙트, 그리고 이러한 인터체인 메시지를 촉진하는 역할을 하는 각 노드에서 실행되는 에이전트를 통해 지원됩니다.

SKALE 체인들은 각각 인박스와 아웃박스를 가지고 있습니다. 다른 체인으로 보내지는 메시지는 아웃박스에 유지되며, 무작위로 지정된 에이전트가 이를 수신하여 적절한 수신 체인의 인박스로 메시지를 보내고, 해당 체인이 거래가 송신 체인의 블록체인에 포함되었음을 검증하는 데 필요한 추가 메타데이터[7]도 제공합니다. 수신 블록체인에서 이것이 확인되면, 거래는 온체인 메시징 프록시를 통해 대상 주소 또는 스마트 컨트랙트로 전달됩니다.

상위 블록체인(예: 이더리움 메인넷)에서의 가치 전송의 경우, *DepositBox*가 현금화 캐싱 메커니즘 및 양방향 페그로 사용되며, 각 SKALE 체인에서 이 풀링된 가치에 대한 바우처가 발행되고 거래와 동일한 방식으로 참가자 간에 자유롭게 교환됩니다. 가치가 SKALE 체인 간에 교환될 때, 이중 지출 공격의 가능성을 제거하기 위해 먼저 송신 체인에서 가치가 소멸되고 수신 체인에서 생성됩니다. 이는 DepositBox에 잠긴 자본을 해제하는 이더리움 메인넷으로 전송된 상환 거래에도 적용됩니다.

[7] 수신 체인은 트랜잭션이 해당 블록에 포함되었는지와 해당 블록이 현재 노드 검증자 세트에 대한 유효한 BLS 서명을 가지고 있는지를 확인하기 위해 네트워크 내의 에이전트는 트랜잭션이 포함된 커밋된 블록을 수신 체인에 반드시 보내야합니다.

Governance

SKALE은 모든 커뮤니티와 SKL 토큰 보유자들의 통제 하에 운영되는 완전한 탈중앙화된 프로젝트입니다. 어떤 기관이나 단체도 네트워크에 대한 통제권이나 권한을 가지고 있지 않습니다. 모든 네트워크 결정은 그룹 내 합의를 통해 이루어집니다. 개발자들은 네트워크 개선 제안을 만들기 위해 노력합니다. 네트워크 경제에 영향을 미치는 모든 개선을 위한 제안은 투표를 통해 승인되어야 합니다. 투표 메커니즘 자체는 유연하며, 제안과 투표에 따라 변경될 수 있습니다. 모든 코드 변경 사항은 검증자 운영자에 의해 독립적으로 승인, 업데이트 및 실행되어야 합니다.

현재의 투표 메커니즘과 제안은 다음에서 확인할 수 있습니다:

- https://snapshot.org/#/skale.eth
- https://forum.skale.network/

SKALE Token - SKL

SKALE 토큰(SKL)은 검증자로서 네트워크에서 작업하거나 개발자로서 일정 기간 동안 SKALE 체인을 배포하고 임대하여 리소스 공유에 액세스할 수 있는 권리를 나타내는 하이브리드 유틸리티 토큰입니다.

사용자는 구독 모델에 의해 SKALE에 비용을 지불하고 리소스(컴퓨테이션, 저장소, 대역폭)를 하나의 SKALE 체인 형태로 미리 정해진 기간 동안 대여할 수 있습니다.

검증인들은 네트워크에 SKALE을 스테이킹 하면, 노드를 운영할 권리를 얻으며, 수수료와 인플레이션을 통한 토큰을 획득합니다.

SKL 토큰 경제에 대한 자세한 내용은 SKALE 네트워크 웹사이트에서 확인할 수 있습니다:
https://skale.space/network#token

Summary

SKALE 네트워크는 블록체인의 대중적인 확장을 달성하기 위해 참신한 “Built Different” 모델을 사용합니다. 전 세계 블록체인 애플리케이션의 개발자와 사용자에게 효율적인 비용으로 가스비 없는 확장성을 제공하는 것이 탈중앙화 프로젝트인 SKALE의 과제입니다.

여러 확장성 기술과 아키텍처가 확장성과 보안을 약속하지만, SKALE의 Built Different 모델은 게임, AI, DePin, 소셜, DeFi 등 실제로 다양한 응용 분야에서 효과적으로 응용 가능함을 입증하였습니다.

SKALE Terminology

Term	Description
ABBA	유효성, 합의 및 확률적 종료를 보장하는 무작위화된 합의 프로토콜.

ABFT	정직한 참여자들 간의 일부 메시지가 지연되거나 의도된 수신자에게 전달되지 않을 가능성을 고려하는 비잔틴 장애 허용의 한 유형.
Catchup Agent	각 노드에 포함되어 SKALE 체인의 현재 '최신 상태'로 노드를 동기화하는 책임을 지는 에이전트입니다. 이는 오류가 발생한 노드가 다시 온라인 상태로 돌아와 놓친 블록을 다운로드하고 검증할 수 있게 하며, 기존 SKALE 체인에 새로 추가된 노드의 동기화를 가능하게 합니다.
DepositBox	SKALE 체인과 이더리움 메인넷 모두에서 자본의 잠금 및 해제를 용이하게 하는 스마트 계약입니다. 이 메커니즘은 가치 이전을 위한 양방향 페그 구현의 기반이 됩니다.
Messaging Agent	각 노드에서 실행되며 연결된 SKALE 체인을 청취하여 체인 간 메시징을 용이하게 하는 에이전트입니다. 이러한 에이전트들은 검열의 가능성을 줄이기 위해 무작위로 지정된 시간에 수신 메시지를 듣도록 임명됩니다.
Parent Blockchain	일방향 또는 양방향 페그를 통해 가치가 이전되는 블록체인입니다.
Peer Node	네트워크에서 특정 노드의 가동 시간과 지연 시간을 모니터링하는 노드들입니다. 피어 노드들은 이러한 메트릭을 정기적인 네트워크 기간에 SKALE 매니저에게 보고하며, 이 메트릭은 해당 노드의 보상을 결정하는 데 사용됩니다.
SKALE Chain	무작위로 선택되고 자주 교체되는 네트워크 노드 세트를 사용하는 지분 증명 기반 체인입니다. 이러한 노드들은 집합적으로, 그러나 탈중앙화되고 독립적인 방식으로 사용자 트랜잭션을 수락하고 SKALE 합의를 실행하며, SKALE 체인의 동일한 사본을 저장합니다. 일반적인 구현은 Docker 컨테이너와 같은 가상화 컨테이너에서 네트워크 노드를 실행하여 단일 물리적 서버가 여러 SKALE 체인을 지원할 수 있게 합니다.
SKALE Manager	이 계약서(이더리움에 위치)는 SKALE 체인의 생성/파괴, 노드 생성/파괴, 출금 및 보상을 포함하여 네트워크 내 모든 엔티티의 오케스트레이션을 관리합니다.
SKALE Super Node	SKALE 네트워크 내의 슈퍼 노드로, 노드의 생성 및 파괴를 위한 자원 오케스트레이션, 네트워크 내 다른 노드의 모니터링 및 SKALE 매니저와의 인터페이스를 담당합니다.
Two-Way Peg	하나의 블록체인에서 다른 블록체인으로, 그리고 그 반대로 디지털 자산을 전송할 수 있게 합니다. 전송 시 한 블록체인에서 디지털 자산을 잠그면서 두 번째 블록체인에서 동등한 양을 잠금 해제합니다. 원래의 디지털 자산은 두 번째 블록체인에서 동일한 양의 토큰이 다시 잠길 때 잠금 해제될 수 있습니다.
Node	노드들은 슈퍼 노드 아래에 할당되어 SKALE 체인에서 네트워크 검증, 데이터 가용성 및 저장 기능을 수행합니다.

SKALE Protocol Parameters

Name	**Description**

BEACON_TIME	빈 블록 생성 간의 시간입니다. 블록체인에 아무도 트랜잭션을 제출하지 않으면 빈 비콘 블록이 생성됩니다. 비콘 블록은 블록체인의 정상적인 작동을 감지하는 데 사용됩니다.
COMMON_COIN	Rabin [7]이 도입한 것으로, 각 프로세스에 동일한 시퀀스의 랜덤 비트 b_1, b_2, ..., b_r, ...를 제공하는 분산 객체입니다.
MAX_BLOCK_SIZE	블록 본문의 최대 크기(바이트 단위)입니다. 현재는 8MB를 사용하며, 미래에는 특정 평균 블록 커밋 시간을 목표로 블록 크기를 자동 조정하는 것을 고려할 수 있습니다.
N_WIN	노드들이 커밋 및 최종화에 유효하다고 합의한 제안의 총 수입니다. 이 숫자는 모듈로 나눗셈에 사용되어 승리한 제안의 인덱스를 결정합니다.
TIP_ID	블록체인에서 최신 블록(또는 "팁")의 ID입니다.

References

1. Achour Mostefaoui , Hamouma Moumen , Michel Raynal, Signature-free asynchronous byzantine consensus with t < n/3 and o(n2) messages, Proceedings of the 2014 ACM symposium on Principles of distributed computing, July 15-18, 2014, Paris, France
2. Alexandra Boldyreva. Threshold Signatures, Multisignatures and Blind Signatures Based on the Gap-Diffie-Hellman-Group Signature Scheme. In Yvo Desmedt, editor, Public Key Cryptography - PKC 2003, volume 2567 of LNCS, pages 31–46. Springer, 2003.
3. P. Feldman. A Practical Scheme for Non-interactive Verifiable Secret Sharing. In FOCS'87, pages 427–437, 1987.
4. Beuchat, J.L., Díaz, J.E.G., Mitsunari, S., Okamoto, E., Rodríguez-Henríquez, F., Teruya, T.: High-speed software implementation of the optimal ate pairing over barreto–naehrig curves. In: Joye, M., Miyaji, A., Otsuka, A. (eds.) Pairing 2010. LNCS, vol. 6487, pp. 21–39. Springer, Heidelberg (2010)
5. Boneh, D., Shacham, H., Lynn, B.: Short Signatures from the Weil Pairing. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 514–532. Springer, Heidelberg (2001)
6. Christian Cachin , Klaus Kursawe , Victor Shoup, Random oracles in constantipole: practical asynchronous Byzantine agreement using cryptography (extended abstract), Proceedings of the nineteenth annual ACM symposium on Principles of distributed computing, p.123-132, July 16-19, 2000, Portland, Oregon, USA [doi>10.1145/343477.343531]
7. M. O. Rabin. Randomized Byzantine Generals. In 34th Annual Symposium on Foundations of Computer Science, Palo Alto California, 3-5 November 1993, pages 403–409. IEEE Computer Society, 1983.