샤더 기술 백서

샤더 - 크로스-체인 분산 스토리지 프로토콜

V1.1

Website: https://sharder.org
Telegram: https://t.me/sharder_talk
Twitter: https://twitter.com/SharderChain
Medium: https://medium.com/@SharderChain
Email: hi@sharder.org
Github: https://github.com/Sharders

foundation@sharder.org

머리말

오늘날의 세계는 데이터 중심의 사회입니다. 정보 기술과 스마트 라이프의 발전은 폭발적인 데이터 증가로 이어졌습니다. 한편으로는 스토리지 용량의 증가가 데이터 증가보다 크게 뒤떨어져 스토리지 공급이 수요를 못 따라가고 있는 반면, 유휴 상태에 있는 개인 및 기업에 속하는 스토리지 공간은 상당히 많아 낭비되고 있습니다. 또한 암호화의 부재, 데이터 유출 및 남용의 가능성, 조작 가능, 비영구성 및 높은 비용과 같이 현재의 중앙화된 스토리지 시스템에는 많은 어려움이 있습니다.

블록체인은 분산 스토리지(distributed storage), 합의 원장(consensus ledger), P2P 전송, 암호화 알고리즘 및 인센티브 메커니즘을 통합하는 새로운 정보 기술입니다. 탈중앙화, 오픈 소스, 자율성, 익명성, 추적성, 불변성과 같은 선천적 속성은 중앙화된 스토리지 시스템의 문제점을 효과적으로 해결합니다. 샤더 프로토콜은 블록체인 3.0 을 기반으로하는 크로스-체인 분산 스토리지 프로토콜로, 분산 저장소부터 시작하여 다른 많은 상업적 어플리케이션들로까지 확대될 수 있도록 현재의 블록체인 기술을 크게 최적화하는 것을 목표로 합니다.

기술 혁신 : 샤더는 크로스-체인 분산 스토리지 프로토콜을 시작하고 네트워크에서 감시자(Watcher) 및 증명자(Prover) 역할을 창의적으로 도입합니다. 비트코인의 UTXO 모델과 호환되는 자체 샤더-UTXO 모델(Sharder-UTXO Model)을 개발합니다. 샤더 프로토콜에서 실행되는 샤더 체인(Sharder Chain)은 견고성, 보안, 개인 정보 보호 및 시스템 가용성과 함께 발전되었습니다.

공유 경제 : 샤더 프로토콜을 기반으로 하는 크로스-체인 분산 스토리지 네트워크는 가입자에게 안전하고 효율적이며 저렴하며 영구적인 저장 서비스를 제공하는 동시에 가입자들의 여분의 저장 공간을 인센티브를 받고 공유할 수 있게 합니다.

샤더 마이너(Sharder Miner) : 샤더는 컴퓨팅 파워와 저장 공간을 네트워크에 공유함으로써 다양한 보상을 얻을 수 있도록 하는 마이크로 노드 마이너(Sharder Hub)와 올인원 스토리지-마이너 (Sharder Box)를 출시할 것입니다.

상용 애플리케이션 : 샤더 프로토콜은 오픈 소스이고 무료이며, 어떠한 퍼블릭 체인이나 스토리지 네트워크라도 샤더 프로토콜을 사용할 수 있으며, 누구나 DApp 을 개발할 수 있습니다. 빈 클라우드(Bean Cloud)는 정부, 은행, 의료, 전자 상거래 등에서 창출되는 방대한 전자 계약에 대한 저장, 인증 및 보안 서비스를 제공하는 최초의 애플리케이션입니다. 샤더는 샤더 매트릭스(Sharder Matrix), 샤더 브레인(Sharder Brain), 원 페어(One Fair) 및 데이터, AI, 거래와 관련된 애플리케이션들을 개발하고 있습니다.

샤더는 사진과 문서와 같은 평범한 데이터 외에도 생물학적 데이터(유전자 정보, 성장 기록, 의료 기록 등)와 사고와 기억까지도 저장하고자 합니다. 우리의 임무이자 비전은 여러분의 이야기를 저장하는 것입니다. 여러분의 지원으로 우리의 꿈이 실현될 것입니다.

운영(Administration) : 개방성, 투명성, 민주주의의 정신으로 샤더 재단은 샤더 프로토콜의 연구 개발, 샤더 커뮤니티 관리, 샤더 제품 및 문화 홍보를 담당합니다.

목차

1. 초록

샤더 프로토콜은 크로스-체인 분산 저장소 프로토콜입니다. "샤더(Sharder)"라는 이름은 컴퓨터 과학의 "조각(shard)"이라는 단어에서 유래했습니다. 샤더 클라이언트는 다양한 퍼블릭 체인, 스토리지 네트워크, 개인 노드 등에 적용될 수 있습니다. 샤더 프로토콜(Sharder Protocol)은 여러 개체(multiple objects), 동작 기능(action functions), 검사 메커니즘(check mechanism), 합의 메커니즘(consensus mechanism), 기여도 정량화(contribution quantification) 및 승인 메커니즘(authorization mechanism) 등을 정의하고 데이터 암호화, 데이터 샤딩, 멀티 체인 아키텍처, 파일 시스템, 스마트 계약, 자유 시장, 보안, 가용성, 유연성 등을 설계합니다.

이 기술 백서를 통해 컴퓨터 과학, 프로그래밍, 수학 또는 블록체인에 대한 배경 지식이 없는 사용자들을 포함한 모든 사용자들이 어떻게 샤더 프로토콜이 탈중앙화 스토리지 네트워크를 구성하는지를 이해하는데 도움이 되기를 바랍니다.

2. 요약

비전 : 글로벌하고 안전하며, 사생활 보호가 되고, 항상 온라인이며, 크로스-체인을 지원하는 분산 저장 네트워크인 샤더 네트워크를 구축하여 사람들이 중요한 데이터를 저장 및 교환하는 방식을 변혁하는 크로스-체인 공유 스토리지 생태계를 만드는 것입니다.

토큰 (SS) : 샤더 프로토콜에 내장된 암호화폐입니다. 총 5 억 SS 가 발행됩니다. 공식 웹사이트 https://sharder.org 에서 자세한 내용을 알아보십시오.

샤더 체인 : 샤더 프로토콜을 사용하는 최초의 상용 블록체인이며, 최초의 샤더 풀(Sharder Pool, Sharder-Pool0) 및 샤더 네트워크의 초석입니다.

샤더 네트워크: 샤더 프로토콜을 사용하는 다양한 샤더 풀들로 구성된 탈중앙화 및 분산화 네트워크입니다. 저렴한 가격에 양질의 데이터 서비스를 제공할 뿐만 아니라, 빈 클라우드(데이터 저장, 보안, 인증), 샤더 메트릭스, 샤더 브레인, 원 페어 같은 수많은 Dapp 들이 샤더 네트워크 안에서 개발됩니다. 샤더 네트워크는 종국적으로 사람들이 귀중한 데이터를 저장하고 교환하는 방식을 바꿀 것입니다.

샤더 마켓 : 스토리지, 데이터 및 디지털 자산이 교환되는 자유 시장입니다.

탈중앙화 어플리케이션(Dapps) : 빈 클라우드 - 데이터 저장, 인증 및 보안. 샤더 메트릭스 : 개인 생물학적 데이터 저장. 샤더 브레인 : 데이터 보안, 데이터 전송, 데이터 분석, 데이터 검색, 데이터 경고와 같은 빅 데이터 서비스. 원 페어 : 불활성(inert) 데이터 및 가치가 효율적이고 안전하게 유통 및 교환될 수 있는 투명하고 공개된 자유로운 P2P 시장.

오픈 소스 : 샤더 프로토콜은 GPLv2.0 에 기반한 오픈 소스 프로젝트입니다. Github 주소는 https://github.com/Sharders 입니다.

3. 설계 철학

불완전한 노드 가정 : 전체 네트워크가 견고하면서도 단일 노드 장애 및 비정기적인 작동정지(downtime)를 허용하는 네트워크 아키텍처를 기반으로 합니다.

소유권 및 프라이버시 : 데이터는 암호화되어 있고 비공개적(private)입니다. 데이터 소유자는 데이터의 완전한 소유권과 접근 권한을 가집니다. 권한이 없는 누구도 데이터에 접근할 수 없습니다.

정량화된 기여도 : 저장 및 시간 증명(PoST, Proof of Storage & Time) 및 복제 증명(PoR, Proof of Replica)과 같은 측정 방법을 기반으로 네트워크의 모든 기여도를 정량화하고 관찰할 수 있습니다.

최종 일관성 : 데이터 개체는 서로 다른 노드에서 서로 다른 상태를 가질 수 있으며 지속적으로 전체 네트워크에 신속하게 수렴할 수 있습니다.

모니터링 및 복구 : 전체 네트워크 가용성 및 데이터 개체 상태를 면밀히 모니터링하고 어느 정도 자발적으로 복구할 수 있습니다.

감사 및 감독 : 데이터 소유자의 동의를 얻어 특정 상황에서 데이터를 모니터링하고 감사할 수 있습니다.

확장가능한 API : 확장성이 뛰어나며 사용자 친화적인 API 입니다.

4. 샤더 프로토콜 (Sharder Protocol)

우선, 샤더 프로토콜은 비용 효율적인 스토리지 공간, 신뢰할 수 있는 데이터 스토리지 및 투명한 온체인 정보를 제공하는 분산 스토리지 네트워크를 구성합니다. 결국 모든

저장 공간, 저장된 데이터 및 디지털 자산이 함께 자유 시장을 형성하게 됩니다. 샤더 네트워크는 기존 스토리지 리소스뿐 아니라 퀀텀(Qtum), 이더리움(Ethereum)과 같은 글로벌 퍼블릭 체인, IPFS, Aliyun, Baidu Cloud 와 같은 스토리지 네트워크 및 하드 디스크 및 클라우드 디스크와 같은 개인 저장소에도 액세스할 수 있습니다. 모든 DApp 는 샤더 체인을 기반으로 무료로 개발될 수 있습니다.

4.1 샤더 프로토콜 개요

- 개방성과 투명성을 옹호합니다.
- 역할, 네트워크, 데이터, 기여도 정량화, 인센티브 및 멀티 체인 메커니즘으로 구성 구성됩니다.
- PoR (Proof of Replica) 및 PoST (Proof of Storage)의 양적 증명들을 정의합니다.
- 데이터 샤딩, 다중 복제 및 데이터 삭제를 채택하여 데이터 보안 및 가용성을 보장합니다.
- 다양한 퍼블릭 체인 및 네트워크들을 연결하고 멀티체인 생태계를 형성하여 데이터와 가치를 유통합니다.
- Sharder-PAIR 및 Sharder-UTXO 의 인증 메커니즘을 초기설정(initialize)하여 기업 또는 감독 기관의 감사 및 감독 요구 사항을 충족시킵니다.

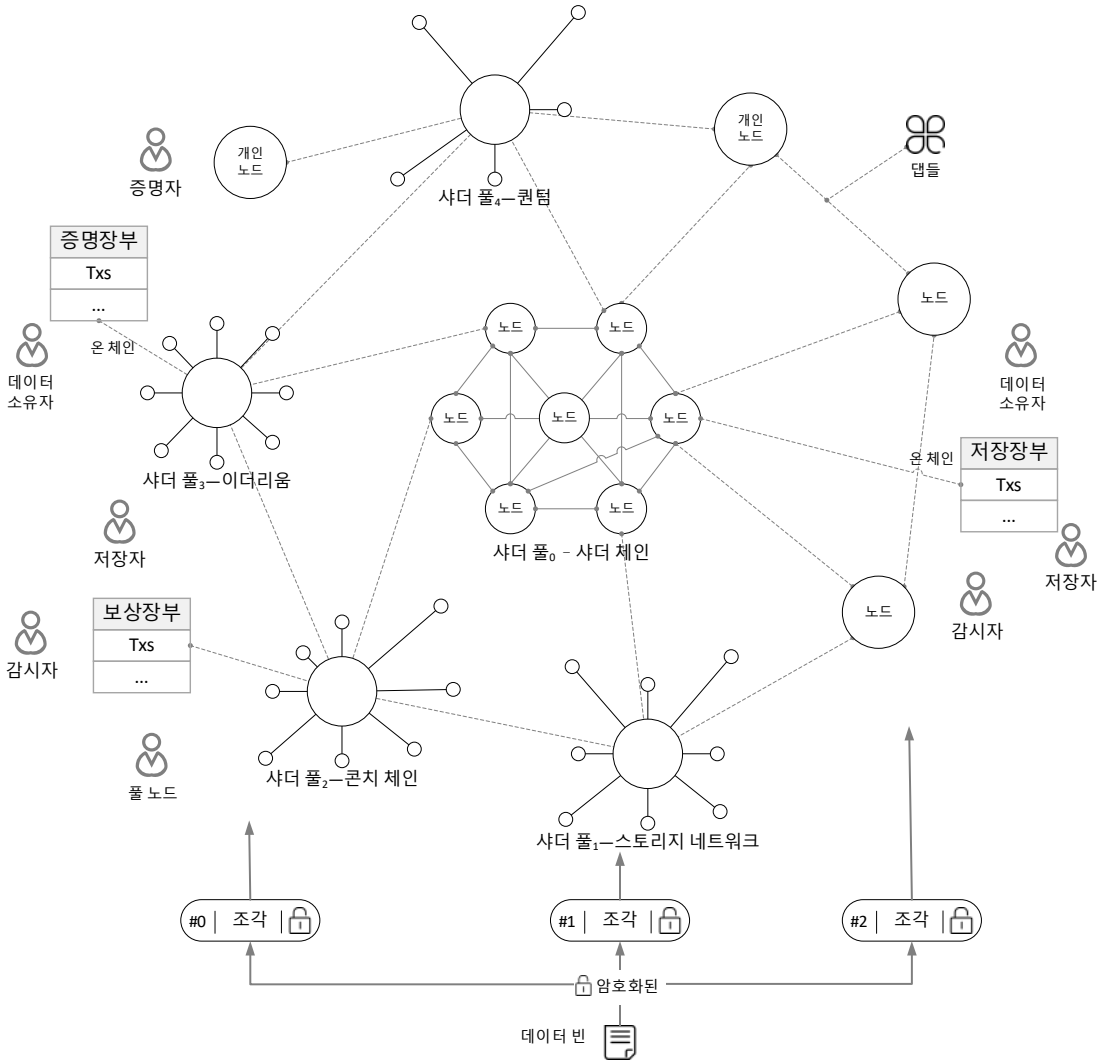


그림 1 샤더 프로토콜 전체 도식도

4.2 역할의 정의

샤더 프로토콜은 네트워크의 모든 참가자의 역할을 정의합니다. 한 노드는 여러 역할을 수행 할 수 있습니다. 풀노드(Full Node)는 신용 증명(PoC, Proof-of-Credit)을기반으로 평가됩니다. [3.8.3 신용 증명]

노드의 종류	채굴자	감시자	저장자	증명자
풀 노드	v	v	v	
저장 노드			v	
감시 노드		v	v	
증명 노드		v		v

표 1 역할과 노드들 간의 관계

빈(Bean) : 샤더 프로토콜에서 데이터 빈은 샤딩 전의 데이터 개체입니다. 샤딩(sharding)을 통하지 않고서는 빈은 나뉘어 질 수 없으며 외부 시스템에서 접근이 불가능합니다.

데이터 소유자 : 데이터 빈의 소유자. 모든 데이터 빈에는 데이터를 저장 노드에 안전하게 저장하기 위해 필요할 때마다 데이터를 확인할 권리가 있는 소유자의 서명이 있습니다.

데이터 소유자는 데이터의 중요성에 따라 다양한 보안 수준을 요구합니다. 이에 따라 샤더 프로토콜은 적절한 보안 전략을 채택하고 작업을 수행할 수있는 적합한 저장 노드를 선택합니다. 물론 보안 수준이 높을수록 스토리지 비용이 높아집니다.

저장자(Storer) : 데이터 저장을 위한 디스크 용량을 제공하고 보상을 얻습니다. Storer 들은 데이터 소유자 또는 감시인(Watcher)으로부터 검사를 받아야하며 저장 증거를 제공합니다. 예를 들어 PoST 는 데이터가 특정 기간 동안 저장되며 언제든지 접근할 수 있음을 증명합니다. 이후 "저장 노드"는 스토리지 클라이언트를 실행하는 물리적 노드를 의미합니다.

감시자(Watcher) : 전체 네트워크 상태를 관찰하고 보안 전략에 따라 보안 상태를 확인하며 현존하거나 잠재적인 허점을 수정합니다. 감시자는 지속적으로 온라인 상태여야 합니다. 이들은 전체 네트워크의 빠른 수렴에 필수적이며 데이터 색인 생성(data indexing)에 완벽해야 합니다.

데이터 소유자를 대신하여 감시자는 저장자에게 비트 디텍션(Beat Detection : 적은 용량의 데이터를 보내고 그에 대한 응답을 확인하는 작업-역자 주)을 무작위로 수행하여 데이터 보안 및 가용성을 확인합니다. 대부분의 작업은 오프체인 (off-chain)으로 수행됩니다. 감시자는 채굴자(Miners)와 저장자를 확인하고 균형을 유지할 수 있는 독립 노드이므로 데이터 보안을 강화하고 적대적인 공격을 피해야 합니다.

채굴자(Miner) : 블록체인 네트워크에 있는 채굴자와 마찬가지로 채굴자는 블록 정보를 저장하고 트랜잭션 및 번들을 처리하기 위해 클라이언트 (터미널 또는 GUI)를 실행해야합니다. 네트워크의 안정성, 연결성 및 처리 속도는 채굴자와 큰 관련이 있습니다. . 일반적으로 풀 노드는 온라인 안정성과 효율성을 보장하기 위해 채굴자 역할을 하게 됩니다. 현재 샤더 체인(Sharder-Pool0)에 연결된 풀 노드 만 PoS 또는 DPoS 를 사용하여 채굴 경쟁을 할 수 있습니다.

증명자(Prover) : 샤더 프로토콜에서 데이터를 디지털 자산으로 변환하고 대중의 신뢰를 데이터에 추가하는 역할을 합니다. 증명자의 증거 데이터(evidence data)와 원래의 데이터 개체(original data objects)가 체인에 기록됩니다. 모든 데이터는 추적 가능하고 조작이 불가능합니다.

대개의 경우 증명자는 외부 기관 또는 공공의 신뢰를 받는 조직입니다. 증명자 노드가 되면 누구나 샤더 네트워크에 참여하여 증명자 역할을 수행할 수 있습니다. 예를 들어, 디지털 저작권에서 가장 권위있는 증명자는 저작권 관리 행정부(National Copyright Administration)일 것입니다. 증명자 노드는 저작권 관리 행정부에 연결되어 샤더 네트워크에서 그 업무를 대신 수행할 수 있습니다. 데이터 인증과 관련해서는 공증인 사무소와 사법 기관은 신뢰성 있는 검증 기관이 됩니다. 공증된 온체인 데이터는 법적으로 유효합니다.

풀 노드(Full-Node) : 뛰어난 대역폭과 처리 능력을 갖춘 안정적인 온라인 물리 노드입니다. 샤더 프로토콜에서 풀 노드는 채굴자, 저장자, 감시자 역할을 할 수 있습니다.

샤더 풀(Sharder Pool) : 샤더 프로토콜을 사용하는 여러 노드들로 구성된 작은 네트워크입니다(채굴풀과 유사함). 퍼블릭 체인 또는 스토리지 네트워크가 샤더 프로토콜을 적용하면 샤더 풀이됩니다. 모든 노드는 샤더 풀에 속해있어야 합니다. 만약 노드가 샤더 풀에 연결되어 있지 않을 경우 샤더 풀 0(Sharder-Pool0)에 속하게 됩니다.

샤더 풀은 다른 샤더 풀들과 연결하지 않고 프라이빗 클라우드 스토리지 네트워크를 구성하는 프라이빗 체인과 같이 격리된 네트워크로 남아있을 수 있습니다. 샤더 프로토콜은 개방성을 추구하기 때문에 미개방된 상태를 유지하고 싶어하는 샤더 풀은 수수료를 지불해야합니다.

샤더 네트워크(Sharder Network) : 샤더 프로토콜을 적용한 풀 노드들은 샤더 네트워크를 형성합니다. 대규모 스토리지 시스템인 샤더 네트워크는 급증하는 기업 및 개인 스토리지 수요를 충족시키기 위해 스토리지에 대한 자유 시장을 구축하는데 전념하고 있습니다. 한편 네트워크는 유휴 또는 오래된 저장 장치를 활용하여 전자 폐기물을 줄이고, 스토리지 공급자에게 보상을 제공합니다.

샤더 체인(Sharder Chain) : 샤더 프로토콜을 적용한 최초의 상용 블록체인 네트워크입니다. 일명 샤더 풀 0(Sharder-Pool0) 입니다. 샤더 체인은 정보 또는 데이터 개체를 영구적으로 기록하는 분산 원장입니다. 샤더 체인은 전통적인 통신 네트워크의 백본

노드와 유사하며, 멀티 체인 샤더 프로토콜에서 중간 고정 네트워크(intermediate anchoring network)로 작동합니다.

장기적으로 샤더 체인은 수많은 분산된 상용 Dapp 들을 통해 자율적이고 일관성 있는 퍼블릭 체인을 구축하기를 원합니다..

샤더 마켓(Sharder Market) : 샤더 네트워크의 탈중앙화 거래 시장입니다. 지난 몇 년 동안 빗쉐어(BitShares)와 이더델타(EtherDelta)는 탈중앙화 거래소가 안정적으로 작동할 수 있음을 증명했습니다. 우리는 샤더 네트워크가 수요 및 공급에 따라 가격이 결정되는 자유 시장으로 진화할 수 있기를 희망하고 있습니다. 샤더 체인은 초기 단계에서 시장 조성(market-making) 및 거래 서비스들을 제공합니다. 점차적으로 판매자와 구매자 스스로가 입찰하게 되고, 샤더는 트랜잭션 기록자의 역할과 참조 가격(reference prices), 과거 가격(historic prices) 및 스마트 계약 서비스를 제공하는 가격 분석가의 역할만을 하게 됩니다. 완전한 블록 정보를 가진 풀 노드는 참조 가격과 시장조성 서비스를 제공할 수 있습니다. 우리는 미래에 테스트, 인증 및 삭제(erasure)와 같은 서비스들이 자유 시장에서 제공될 수 있기를 희망합니다.

4.3 네트워크 형상(Network Topology)

항상 수많은 노드가 생성되고 없어지는 P2P 네트워크를 구축하기 위해서는 라우팅 테이블(routing table) 유지관리 및 검색에 적합한 알고리즘을 갖추는 것이 중요합니다. Kademlia 프로토콜 (이하 Kad) [1]이 우리의 우선 순위이며, 이를 바탕으로 우리는 P2P 네트워크 (Chord 알고리즘을 백업으로 사용)를 구축했습니다. 분산 해시 테이블(Distiributed Hash Table)은 Kad 로 만들어졌으며, XOR 을 기반으로 합니다. 거리를 측정하고 라우팅 검색 프로세스의 속도를 크게 향상시킵니다. 이는 수많은 저장 노드가 있는 샤더 네트워크에 매우 중요합니다. Kad 네트워크를 구현하는 데는 두 단계가 필요합니다. 먼저 간단한 라우팅 테이블을 기반으로하는 P2P 네트워크를 구축한 다음, 저장 노드 클라이언트를 열면서 Kad 네트워크를 개발합니다.

Kad 에서 K 버킷(bucket)의 노드 목록은 노드의 온라인 상태를 나타냅니다. 미래에는 PoC 알고리즘을 사용하여 신용도를 평가합니다. 이 알고리즘은 감시자가 가장 인접한 노드를 선택하는데 도움을 주는 순위의 가중치(weight of ranking)로 사용됩니다.

4.4 데이터 개체 작업 (Data Object Operation)

PRCBean = (Put,Get,Watch)

- Put (data) → key : 클라이언트가 Put 프로토콜을 실행하여 "key"를 고유한 데이터 태그로 사용하는 데이터를 저장합니다.
- Get (key) → data : 클라이언트 실행 Get 프로토콜을 사용하여 "key"라는 고유 태그가 있는 데이터를 가져옵니다.
- Watch () : 감시자는 감시 프로토콜(Watch Protocol)을 실행하여 저장된 데이터를 확인하고, 데이터 개체의 전체 네트워크 상태를 동기화하며, 다양한 보안 전략에 따라 누락된 데이터, 데이터 오류, 비가용성(unavailability) 등의 오류를 수정합니다.

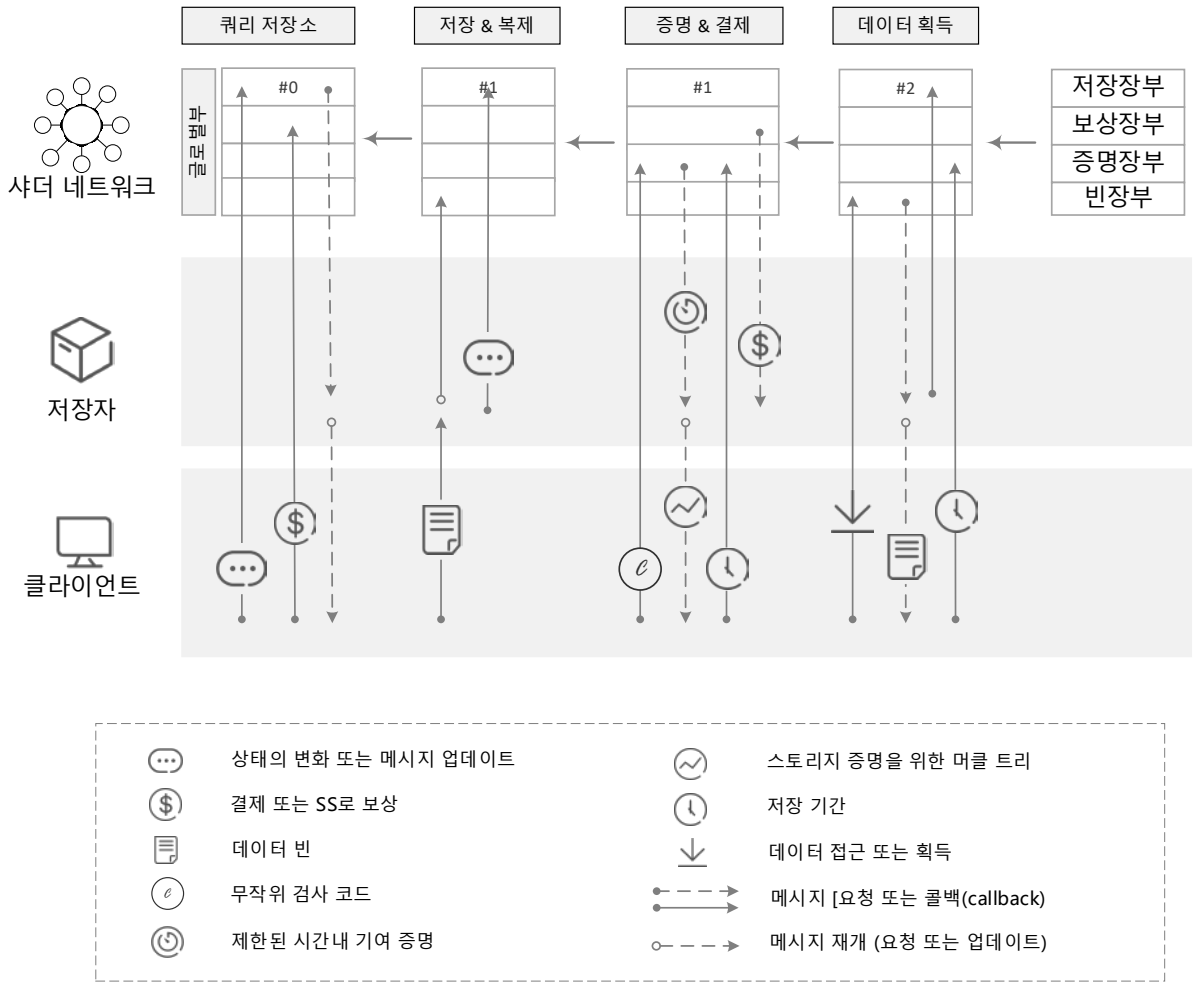


그림 2 데이터 개체 작업

4.4.1 데이터 저장

a. 클라이언트는 저장장부(Store-Book)에 데이터 저장을 요청합니다.
b. 클라이언트는 저장 비용을 지불하고 샤더 프로토콜은 저장자를 매칭시켜 줍니다.
c. 클라이언트가 파일을 저장자에 업로드합니다.
d. 저장자는 데이터를 저장하고 저장장부 및 빈장부(Bean-Book)의 전역 상태(global

state)를 업데이트합니다.

e. 저장자는 보안 전략에 따라 복제 임무(Replica-Task)를 수행하라고 네트워크에 전파합니다.

f. 다른 저장자가 복제본을 만들고 보안 전략에 의해 정의된 수량 요구사항(quantity requirment)을 충족하는지 확인합니다. 그렇지 않은 경우, 저장자는 복제 임무를 수행하라고 계속해서 네트워크에 퍼트립니다.

4.4.2 데이터 획득 (Data Retrieval)

1. 클라이언트는 데이터를 요청하고, 샤더 프로토콜은 빈장부(Bean-Book)에서 최신 데이터
   개체를 가져와서 클라이언트에게 제공하며, 저장자들에게는 동기화를 요청합니다.
2. 활성 모드(active mode)에서 클라이언트는 저장자에 연결하여 데이터를 가져옵니다.
   피동 모드(passive mode)에서는 저장자는 데이터를 클라이언트로 푸시(push)합니다.
3. 저장자는 데이터 획득 후 저장장부를 갱신합니다.
4. 저장자는 데이터 획득 후 저장장부 및 빈장부(Bean-Book)의 전역 상태를 갱신합니다.
5. 클라이언트는 데이터 획득 후 증명장부(Proof-Book)를 업데이트하여 저장자가 데이터 개체를 저장했음을 증명합니다.

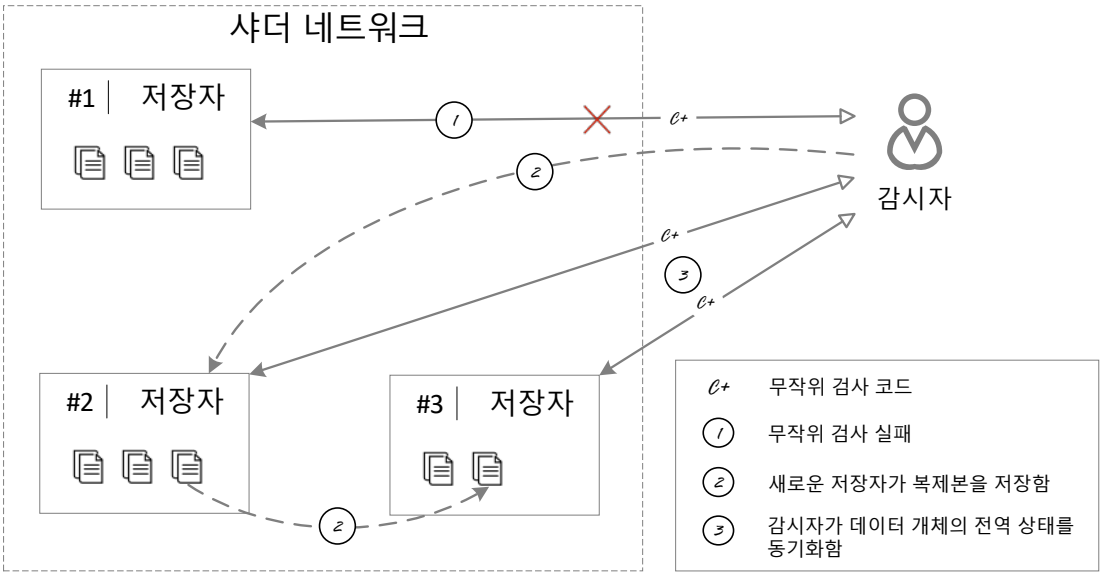


그림 3 감시자는 데이터 스토리지를 검사하고 조정함

복제본 조정 : 만약 한 저장자가 가용하지 않은 경우 감시자는 다른 노드에서 복제본을 만들도록 요구할 수 있습니다. 오리지널 저장자가 일시적으로 오프라인이거나 예기치 않은 가동중지가 발생하는 경우가 있습니다. 문제는 온라인으로 돌아왔을 때 어느

노드가 데이터를 저장하고 보상을 받을 자격이 있는가 하는 것입니다. 샤더 프로토콜은 노드의 신용으로 결정합니다. 일반적으로 더 높은 신용을 가진 저장자는 데이터를 저장하고 보상을 받지만 더 낮은 신용을 가진 저장자는 데이터를 삭제해야 합니다.

리소스 접근 : 데이터 조각(shard)은 글로벌 네트워크에 무작위로 저장되고 복제됩니다. 데이터 입수의 첫 번째 단계는 데이터를 가져올 노드를 아는 것입니다. 즉, 데이터 색인(indexing)이 필요합니다. 감시자는 데이터 상태를 관찰하고 데이터 분배를 조정하여 전체 네트워크 상태를 빠르게 수렴시키는 이상적인 색인 서비스 제공자입니다. 따라서 감시자는 사용자에게 데이터 개체의 최신 주소를 제공할 수 있습니다.

4.4.3 데이터 검사 (Data Check)

1. 클라이언트 또는 감시자는 검사 코드 C 를 무작위로 생성하고 증명장부(Proof-Book)에 기록합니다.
2. 샤더 프로토콜은 검사 코드 C 에 대한 응답으로 저장자가 저장 증명 M(Proof of storage M)을 생성하도록 요구합니다.
3. 저장자는 제한된 시간 내에 클라이언트 또는 감시자에게 저장 증명 M 을 제공합니다.
4. 검사가 완료된 후 클라이언트 또는 감시자는 증명장부를 업데이트합니다.
5. 검사 후 샤더 프로토콜은 보상장부(Reward-Book)를 생성하고 보상의 일부를 저장자에게 지급합니다. 샤더 프로토콜은 머클트리(merkle tree)[6]와 zk-SNARK[7]를 사용하여 저장자가 스토리지를 증명하게 합니다. 저장소에 대한 무작위 검사는 데이터 소유자 또는 감시자에 의해 시작됩니다.
6. PoR [3.8.1 복제본 증명] 및 PoST [3.8.2 저장 및 시간 증명]를 참조하십시오. 감시자는 [3.6 데이터 가용성] 보안 전략에 따라 전체 네트워크 데이터 개체를 정기적으로 검사하고 전체 네트워크 일관성을 유지해야하며 현존하거나 잠재적 보안 혹은 가용성 문제를 해결해야합니다. (예 : ① 조각(shard)이 없거나 사용할 수없는 경우, ② 저장자가 오랫동안 사용할 수 없는 상태이거나 문제가 있을 경우)

4.4.4 상태 수렴 (State Convergence)

데이터 복제를 위한 시간 소비, 데이터 분배에 대한 감시자의 확인 및 조정은 데이터 개체 수렴 검증(data object convergence proof)의 문제로 볼 수 있습니다. 증명은 다음과 같습니다 :
네트워크에 N 개의 노드가 있고 데이터 사본을 저장할 시간이 St 라고 가정합니다. 극단적인 상황에서 N-1 개의 쿼리(query) 후 네트워크의 마지막 노드가 응답하고 사용 가능한 것으로 인식됩니다. 복제의 시간 복잡도는 O(N)+St 입니다. St 는 네트워크가 안정적일 때 상수이기 때문에 시간 복잡도는 O(N) 입니다. 즉, 이용 가능한 노드를 찾기 위한 시간 소비로 단순화 될 수 있습니다. 빈번하게 들어오고 나가는 노드가 있는 네트워크의 O(N)은 수렴되지 않을 것입니다(intolerable). 그러나 Kad 네트워크에 k-bucket 을 도입하면 사용 가능한 노드 조회에 소요되는 시간을 줄일 수 있습니다. 조회해야하는 목표 노드가 t 라고 가정합시다. 모든 쿼리가 t 에 가까운 k- bucket 에서 정보를 얻을 수 있기 때문에, 모든 회귀 연산 (recursion operation)은 거리의 절반 이상을 절약하고 쿼리는 O(logN)의 속도로 빠르게 수렴할 수 있습니다.

4.5 데이터 보안

데이터 암호화 : 저장자에 저장되기 전에 데이터 파일들은 클라이언트에 암호화 (AES-256-CTR)됩니다. 즉, 저장자는 파일 컨텐츠에 액세스할 수 없습니다. 기밀 데이터의 경우 소유자는 샤더 네트워크에 저장하기 전에 데이터를 하드웨어로 암호화할 수 있습니다.

데이터 빈 샤딩(Data Bean Sharding) :

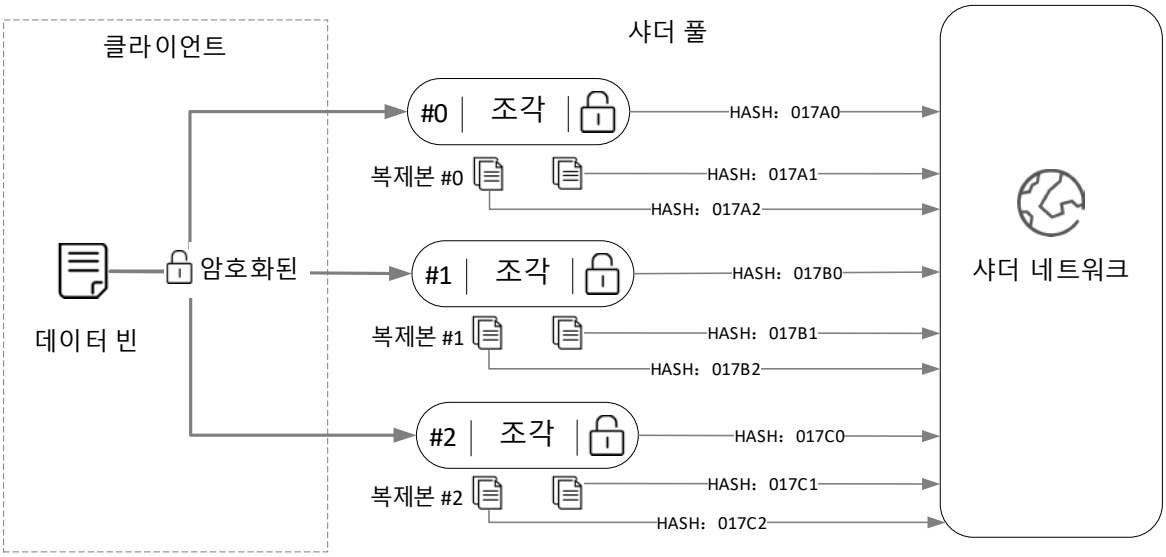


그림 4 데이터 빈 샤딩

데이터 빈 샤딩(이하 샤딩으로 약칭) 전략은 보안 전략과 매우 관련이 있습니다. 데이터 소유자가 높은 데이터 보안을 요구하면 샤딩이 큰 도움이 될 것입니다. 샤딩의 가용성을 보장하기 위해 우리는 삭제 기능(erasure)을 도입합니다.

우리는 저장자가 매우 작은 파일에 대해 속임수를 쓰지 않을 것이라고 생각합니다. 저장자가 데이터 파일을 삭제하고 R 증명을 유지하면 큰 인센티브를 얻을 수 없습니다. 대부분의 경우 대역폭 및 디스크 I/O 로 인해 일반적인 저장자의 성능 병목 현상이 일어납니다. 이는 샤딩 파일이 저장자의 디스크 공간을 가득 채우지 않는다는 것을 말합니다. 그러나 작은 파일이 너무 많으면 액세스 지연이 발생할 수 있지만 샤더 파일 시스템 [3.17.1 샤더 파일 시스템]의 고성능 동시 데이터 처리로 해결할 수 있습니다.

다중 복제본(Multi-replica) : 사더 네트워크에 b 개의 저장자가 있고, 데이터 빈은 p 샤드로 분할되고 각각에는 n 개의 복제본이 있다고 가정합시다. 성공적인 검색 확률 Rs 는 다음과 같습니다.

$$R_S(\mathrm{b},\mathrm{p},\mathrm{n}) = \frac{\binom{b-p}{n-p}}{\binom{b}{n}}$$

b : 샤더 네트워크의 저장자 수
p : 샤드의 수량
n : 복제본의 수량

코드 클립

```
double fac(int p){
    return p == 0 ? 1  : approximation(p * fac(p-1));
}

double choose(int h,int k){
    return fac(h) / fac(k) / fac(h-k);
}

double rs(int b,int p,int r){
    return choose(b-p,r-p) / choose(b,r);
}

double retrieve(int boxerCount, int pieceCount, int replicaCount) {
    return rs(boxerCount,pieceCount,replicaCount);
}
...
```

검색 확률(Retrieval Probability)

저장자	조각	복제본	데이터
100	10	10	5.776904234533874E-14
100	10	50	5.9341967258582 87E-4
200	10	50	3.7276043023296E16
200	50	90	5.7872010853195E44
300	80	90	4.0942349109395 96E131
500	50	200	3.1464595213037 54E45
...			

보안 전략 : 기본 보안 전략은 보통의 재해 복구 계획과 비슷합니다. 하나는 데이터 복사본에 대한 3 개의 복제본을 생성하는 것입니다 : 하나는 해당 노드 혹은 인접 노드에, 하나는 조금 떨어져 있는 노드에, 다른 하나는 해외 노드에. 그러나 높은 수준의 보안 전략에는 더 많은 저장 공간이 필요하고, 보다 복잡한 "감시"및 "조정"이 필요합니다. 샤더 프로토콜을 사용하면 데이터 소유자가 자신의 필요에 따라 보안 전략을 정할 수 있습니다. 현재 설정 가능한 매개 변수는 복제본 수량 및 샤드 수량입니다. 보안 전략은 감시자가 손실된 데이터 문제를 해결하는 방법에 직접적인 영향을 주며, 전체 네트워크 데이터 개체의 수렴 속도에도 영향을 미칩니다.

4.6 데이터 가용성(Data Availability)

데이터 이레이저(Data Erasure) : 이레이저 코드 (Erasure Coding, EC) [2]는 데이터를 조각들로 나누고, 중복 블록을 확장 및 번호를 지정하고, 이를 하드디스크, 저장 노드 또는 기타 물리적인 장치와 같은 다른 위치에 저장하여 데이터를 보호하는 방법입니다. 샤더 네트워크는 저장 공간을 많이 차지하지 않고 데이터 가용성을 보장하기 위해(저장자의 활용도를 높이기 위해) 빈 샤드(Bean Shard)에 데이터 이레이저를 도입하였습니다.

리드 솔로몬 코드 (Reed-Solomon Code, 이하 RS 코드로 약칭)는 자주 사용되는 이레이저 코드로, 매개 변수 n 과 m 이 RS(n, m)로 표기되며, n 은 원래 블록 수량, m 은 검사 블록 수량을 나타냅니다. 전체 복제와 RS 이레이저 코드의 비교는 다음과 같습니다(자세한 알고리즘은 [4]와 [5]를 참조하십시오. 여기서는 자세히 다루지 않습니다).

종류	디스크 활용도	컴퓨팅 소비	네트워크 소비	복구 효율
완전 복제 (3 복제본)	1/3	매우 낮음	낮음	높음
RS 이레이저 코드	n/(n+m)	꽤 높음	높음	낮음

배포 조정(Distribution Adjustment) : 감사자는 현재 데이터 파일이 안전하고 적어도 하나의 리소스에 액세스할 수 있도록 데이터 복제 및 배포를 지속적으로 조정합니다.

4.7 합의 및 블록 생성

비트코인 네트워크의 PoW 합의는 간결하고 명시적인 경제적 인센티브와 합의 메커니즘을 제시하며, 비호스트 분산 네트워크(non-host distributed network)가 잘 작동할 수 있음을 증명했습니다. 그러나 블록 생성 권한을 놓고 경쟁하기 위해 값 비싼 하드웨어를 사용하고 막대한 양의 전력과 컴퓨팅 파워를 소비하는 것은 자원 낭비라고 생각합니다. 또한 하드웨어 "무기 경쟁"으로 인해 많은 양의 전자 폐기물이 발생됩니다. 우리는 합의 블록 생성이 네트워크를 보호하면서 컴퓨팅 파워를 최대한 활용하기를 희망합니다.

합의 블록 생성(Consensus Block Generation) : 멀티체인 합의 블록 생성의 핵심 요소에는 아래 그림과 같이 Tx 번들(Tx-Bundle) 및 샤더 블록(Sharder Block)이 포함됩니다. 이 방법을 사용하면 각 샤더 풀이 자체 내부 컨센서스와 많은 트랜잭션 기록을 담고있는 블록인 Tx 번들을 가질 수 있습니다. 궁극적으로 풀 노드(Full Node)는 샤더 블록을 생성하여 샤더 네트워크에 전파합니다. 각 Tx 번들은 노드 ID(Node-ID), 풀 ID(Pool-ID) 및 지역 ID(Area-ID)를 포함한 데이터 정보를 기록합니다.

풀 노드(Full Node)는 하나의 샤더 풀에만 연결됩니다. 샤더 블록을 연결하려면 노드가 샤더 체인(Sharder-Pool0)에 연결되어야 합니다. 우리는 샤더 풀이 자체적으로 블록을 연결할 수 있게 하려고 합니다. 이를 구현하기 위한 방법은 각 샤더 풀 안의 샤더 체인에 연결하는 하나 이상의 프록시 노드 (Sharder Agent)를 적용하는 것입니다.

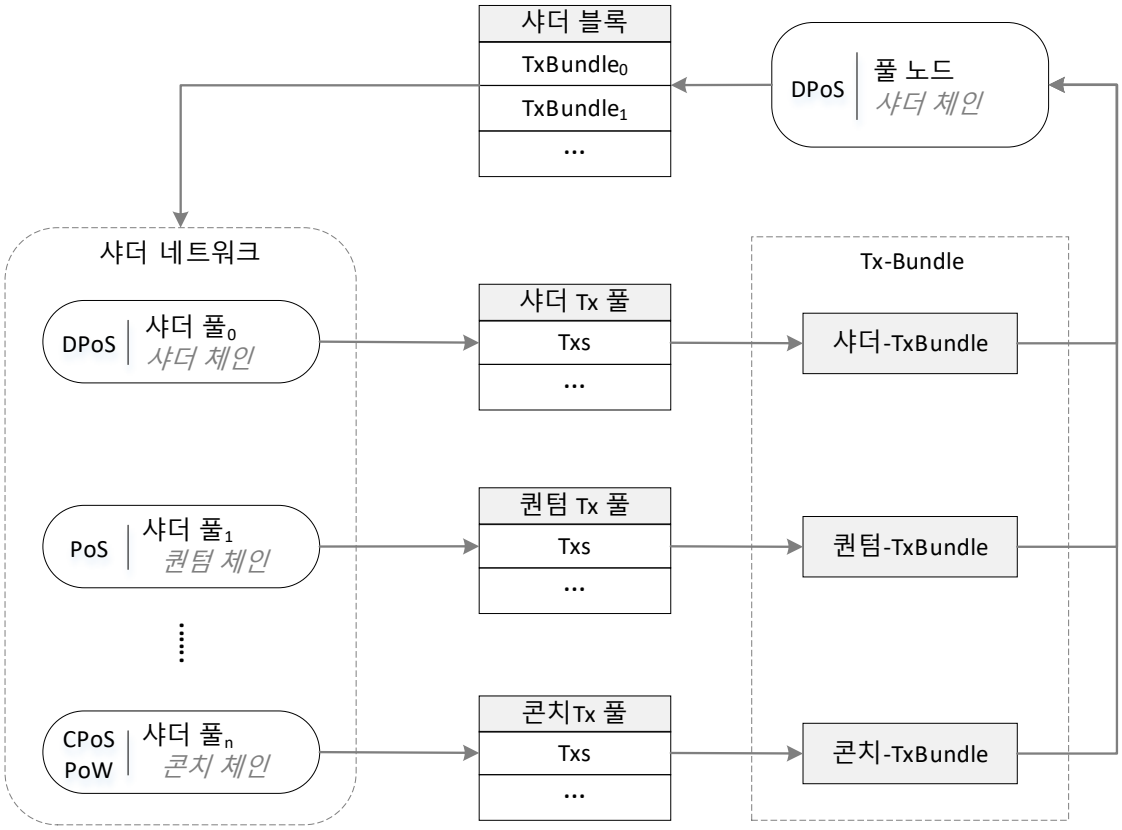


그림 5 멀티 체인 컨센서스

온체인 정보(Information On-chain) : 모든 데이터와 정보가 온체인일 필요는 없습니다. 실제로 샤더 네트워크의 대부분의 데이터는 온체인이지 않습니다. 예를 들어, 데이터 파일은 체인에 저장되지 않습니다. 온체인 URI 는 데이터 개체의 사용 가능한 리소스 주소를 가리키는 표시기(indicator)로 작동합니다.

기본 블록 정보 외에도 다른 온체인 정보에는 회계 트랜잭션, 데이터 개체, 스토리지 트랜잭션, 인증 트랜잭션 등이 포함됩니다. 하나의 스토리지 트랜잭션이 하나의 데이터 개체에 상응하지만 하나 또는 여러 개의 보상 트랜잭션을 일으킬 수 있습니다(PoST 에 근거한 스토리지 보상).

4.8 기여도 정량화(Contribution Quantification)

저장자의 기여도를 증명하는 PoR (Proof-of-Replica) 및 PoST (Proof-of-Storage & Time)를 생성하기 위해 머클 트리 [6] 및 zk-SNARK [7]가 도입되었습니다. 신뢰할 수있는 저장자는 짧은 시간 내에 PoR 로 복제본이 있음을 증명할 수 있습니다. 낮은 신용을 가진 저장자는 저장 및 시간의 증명을 제공하기 위해 PoST 를 채택해야 합니다.

4.8.1 복제 증명(Proof-of-Replica)

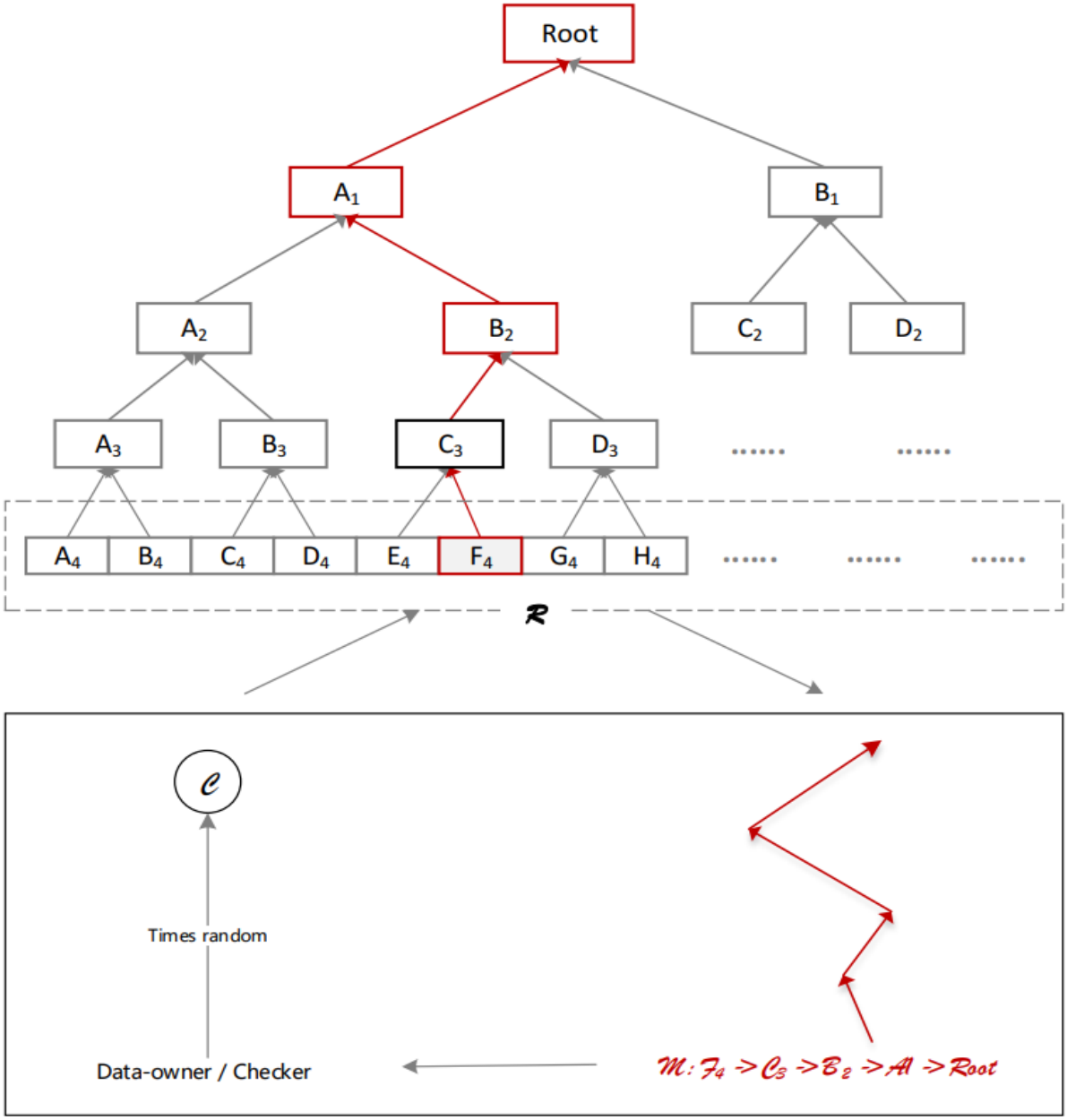


Figure 6 PoR Flowchart

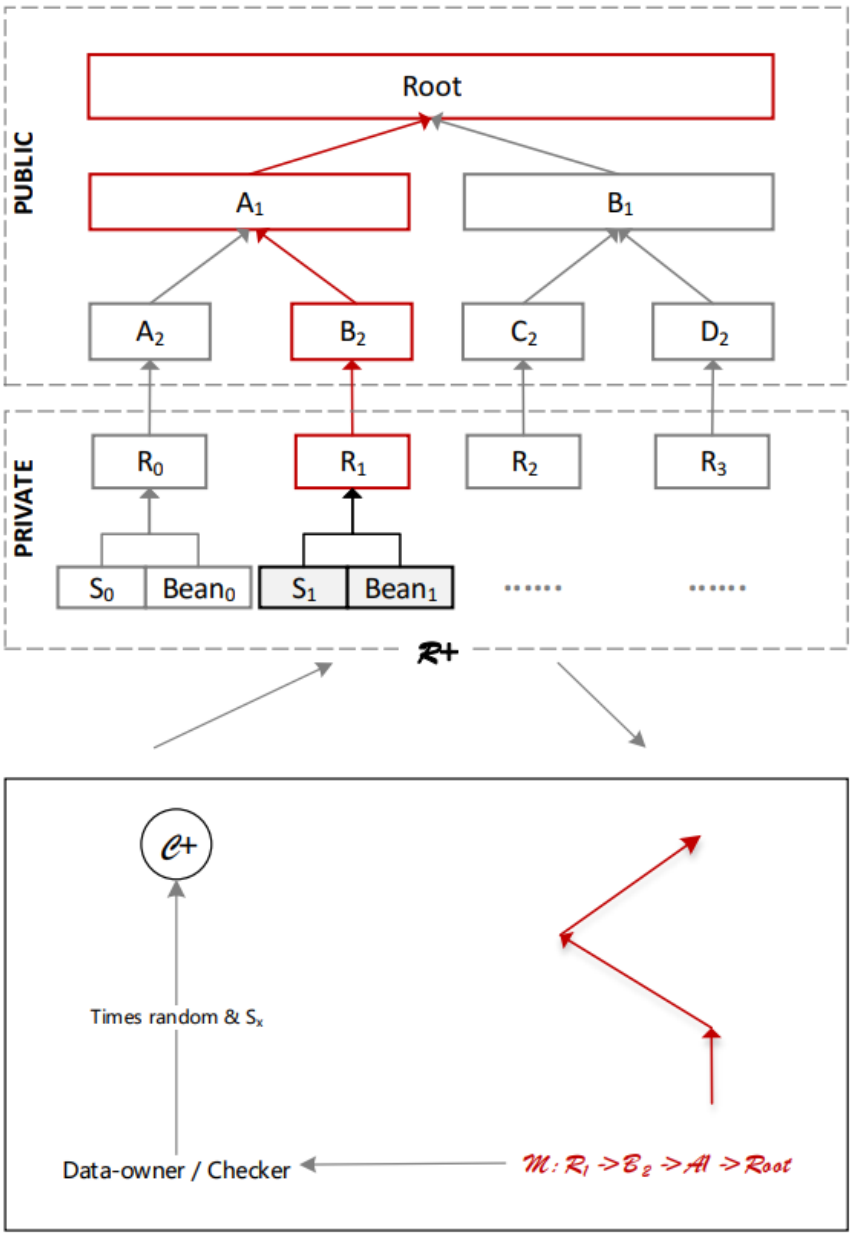
그림 6 PoR 흐름도

저장자가 샤더 네트워크에서 데이터 복제본을 보관할 수 있도록 데이터 소유자는 수시로 다음을 요청할 수 있습니다.

- 데이터 소유자가 시간에 따라 검사 코드 C 를 생성하여 샤더 네트워크로 보냅니다.

- 저장자는 C 에 따라 데이터 조각(shard)을 찾아 $\overrightarrow{M}$ (Merkle tree)를 만듭니다.
- 검사가 통과되면, 샤더 네트워크는 저장장부(Store-Book) 및 보상장부 (Reward-Book)를 업데이트하여 저장자에게 보상금을 지급합니다.

4.8.2 ST 증명 (저장 및 시간)



C+	무작위 검사 코드, 데이터 소유자 혹은 검사자에 의해 생성됨
Beanx	데이터 빈(bean) 조각, 저장자가 저장함
Sx	엔트로피 시퀀스(entropy sequence), 데이터 소유자가 생성함

$\mathbf{R}_x$	머클 트리의 Pre-code, $\mathbf{Bean}_x$ 및 $\mathbf{S}_x$로부터 생성된 저장자
$\mathbf{M}$	머클 트리, 저장자가 생성함, 데이터 소유자 혹은 검사자가 검사함

그림 6 PoST 흐름도

PoR 은 저장자가 데이터를 저장하고 복제본을 만들도록 보장하지만 악의적인 저장자가 부정 행위를 하는 것을 막지는 못합니다. 이런 경우를 생각해 봅시다 :

1. 저장자가 데이터를 복제하고 모든 샤드와 분리 시퀀스(split squence)에 대한 머클 검사
   수(Merkle check number)를 계산한 후, 샤딩 파일을 삭제하고 머클 트리만 보관합니다.
2. 저장자가 증명 요청(request for proof)을 받으면 다른 복제 노드들한테 데이터를 요청하고 C 에 대한 머클 트리를 계산합니다.

따라서 우리는 저장자가 샤딩 파일을 삭제하면 머클 트리를 계산할 수 없고 검사를 통과할 수 없도록 PoST 의 업그레이된 증명을 도입합니다.

- 데이터 소유자는 데이터 샤딩 후 엔트로피 시퀀스 S(entropy squence S)를 생성한 다음 이를 데이터 조각과 함께 사용하여 해시 값 R 을 생성합니다.
- 데이터 소유자가 샤더 네트워크에 때때로 PoST 를 요청하는 S_x (시간에 따른 엔트로피 값, 반복 할 수 없음)를 보냅니다. 그런 다음 저장자는 S_x 및 데이터 조각을 기반으로 Rx 를 계산하고, 나아가 Rx 를 기반으로 머클 트리를 생성합니다.

감시자는 엔트로피 시퀀스 S 를 사용하여 데이터 소유자를 대신해 데이터를 확인할 수 있습니다. 데이터 소유자는 감시자에게 엔트로피 시퀀스의 일부를 제공할 수 있으며 후에 PoST 를 실행할 수 있습니다. 이러한 종류의 프록시는 스마트 계약으로도 실현될 수 있습니다.

4.8.3 신용 증명 (Proof-of-Credit, PoC)

샤더 프로토콜에서 PoC 는 계정에 연동되어 있습니다. CPOS (ConchChain Proof of Stake)에서 파생된 샤더 PoC 공식의 매개 변수(parameter)는 역할에 따라 다릅니다.

- 저장자(Storer) : 총 저장 용량, 저장 기간, 온라인 지속시간, 페널티 정도.
- 풀 노드(Full Node) : 최대 트랜잭션 처리, 블록 생성 속도, 포크 수렴 속도(fork convergence speed), 온라인 지속시간.
- 감시자(Watcher) : 색인 서비스 성능, 온라인 지속시간
- 데이터 소유자(Data Owner) : 데이터 스토리지 용량, 트랜잭션 용량

- 증명자(Prover) : 증명 용량

4.9 인센티브

4.9.1 시스템 보상

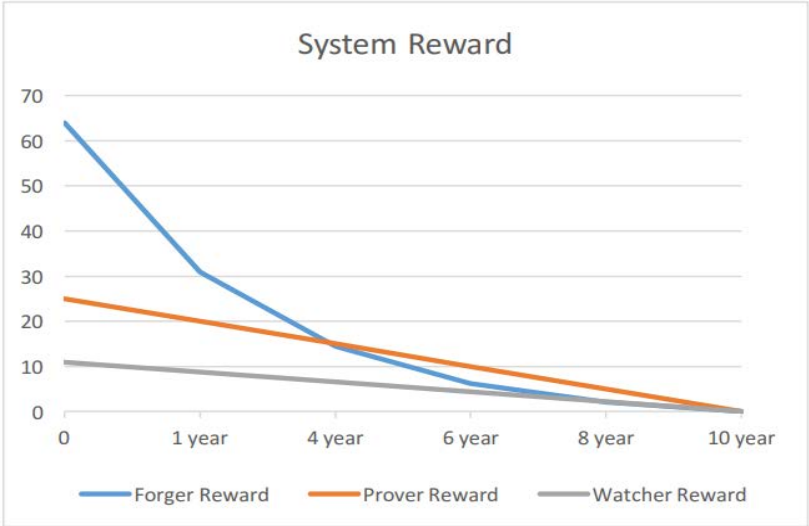


Figure 8 Reward Distribution
그림 8 보상 분배

더 많은 노드가 네트워크에 참여하도록 유도하고 보다 안전하고 강력한 시스템을 구축하기 위해 샤더는 네트워크에 기여하는 노드를 보상합니다.

블록 생성자 보상 : 블록 생성 시간은 5 분 (블록 생성 시간이 10 초로 빨라질 수 있음)으로 설정되므로 하루 288 블록, 연간 105,120 블록이 생성됩니다. 블록 생성자는 블록 당 218 SS 의 보상을 얻습니다. 보상은 1,208,160 개 블록이 생성 될 때까지 210,240 개 블록마다 반으로 줄어들며, 그 후 보상은 블록 당 14SS 로 유지됩니다. 10 년 이내에 64,000,000 SS 가 보상될 것으로 추정됩니다. 모든 보상을 받은 후에는 스토리지 트랜잭션 서비스 수수료 및 기술 서비스 수수료가 네트워크 유지에 대한 경제적 인센티브로 사용됩니다.

감시자 보상 : 샤더 네트워크의 초기 단계에서 공식 노드들은 감시자로 활동합니다. 전체 네트워크가 안정화 상태가 되면 풀 노드가 감시자와 경쟁하게 됩니다. 이 시스템에는 감시자의 보상을 위해 25,000,000 SS 가 준비되어 있는데, 이는 감사자의 PoC 를 기반으로 계산되어 스마트 계약으로 감시자에게 자동 분배됩니다.

증명자 보상 : 샤더 네트워크는 공인된 기관이나 연구소가 증명자로 일하고 자산증명 (PoA, Proof of Asset) 서비스를 제공하도록 장려하기 위해 증명자의 보상으로 11,000,000

SS 를 준비해놓았습니다. 또한 샤더는 데이터 소유자 및 증명자가 서비스 가격을 자유롭게 설정할 수 있도록 합니다.

4.9.2 시스템 처벌

샤더 네트워크는 다음과 같은 행위들에 대해 샤더 토큰을 압수하고 노드의 신용을 낮춤으로써 네트워크에 해를 끼치는 행위를 처벌합니다.

데이터 손실 : 데이터를 잃어버린 노드는 향후 데이터 저장에 대한 모든 보상에서 제외되며 신용이 낮아집니다. 신용이 임계값 아래로 떨어지면 노드가 블랙리스트에 올라서 샤더 네트워크에서 영구적으로 추방됩니다.

악의적인 공격 : 블록 생성을 악의적으로 거부하는 것, SS 에 대한 부정 행위, 샤더 네트워크를 공격하는 것과 같은 행위는 엄중한 처벌을 받게 됩니다. 이런 노드 및 관련 사용자는 블랙리스트에 올라서 샤더 네트워크에서 영구적으로 추방되며 노드의 계정에 있는 모든 SS 는 몰수됩니다.

사기 : 일반적으로 사기 행위가 발생한 후에 사기가 감지됩니다. 따라서 손실을 복구하는 효과적인 방법은 현재 없습니다. 현재의 벌칙은 사용자의 신용을 낮추는 것입니다. 신용이 임계값보다 낮아지면 사용자는 블랙리스트에 올라서 샤더 네트워크에서 영구적으로 추방됩니다. 사기를 막는 가장 좋은 방법은 사기 행위에 대한 비용을 증가시키는 것입니다. 어떤 경우에는 노드들에게서 보증금(deposit)을 요구할 수 있습니다.

4.9.3 트랜잭션 보상(Transaction Reward)

샤더 자유 시장(Sharder Free Market)에 기여한 것에 대한 보상.

저장 보상 : 저장 공간을 제공하는 저장자는 데이터 소유자로부터 보상을 받습니다. 가격은 시장에 따라 결정됩니다.

서비스 보상 : 샤더의 자유 시장이 충분히 활발 해지면 장래에 노드는 맞춤형 서비스 (예 : 독립적인 데이터 색인 서비스, 사용자정의 라이트 지갑 클라이언트 등)를 제공함으로써 보상을 얻을 수 있습니다.

4.10 샤더 토큰 (SS)

샤더(SS)는 샤더 프로토콜에 내재된 암호화폐입니다. 샤더 생태계에서 네트워크에 기여하는 노드에 대한 보상, 악의적인 노드들을 처벌, 그리고 발생 가능한 무한 루프 논리 폭탄(infite-loop logic bomb)을 피할 수있는 인센티브로 사용됩니다. 또한 SS 는 샤더의 멀티 체인 아키텍처에서 고정 토큰(anchoring token)으로 작동합니다.

4.11 스마트 계약 (Smart Contract)

스마트 계약은 이더리움에 의해 신뢰할 수 있고 효율적이라는 것이 입증되었습니다. 샤더 프로토콜의 스마트 계약은 두 단계로 실현됩니다. 1 단계는 트랜잭션 모델의 상위 레벨을 지원하는 인센티브로 샤더 토큰을 채택하는 튜링완전하지 않은(non-Turing-complete) 스마트 계약입니다. 2 단계에서는 가상 머신(virtual machines), 튜링완전(Turing-complete), 오라클(Oracle), 해시 잠금(Hash lock) 등과 같은 고급 기능이 추가됩니다.

단계 I 스마트 계약은 고전적인 FILO 스택 구조를 채택합니다. 원자 연산의 기본 정의는 OP_INIT, OP_EMPTY, OP_FULL, OP_PUSH, OP_POP, OP_DUP, OP_COUNT, OP_HAS, OP_PROOF, OP_CHECKSIGN, OP_EQUAL 입니다. 어플리케이션이 늘어남에 따라 더 복잡한 조작을 실현하기 위해 더 많은 원자 조작 문자(atomic operation characters)가 활용됩니다. 보상을 위한 스마트 계약의 클립은 다음과 같습니다 :

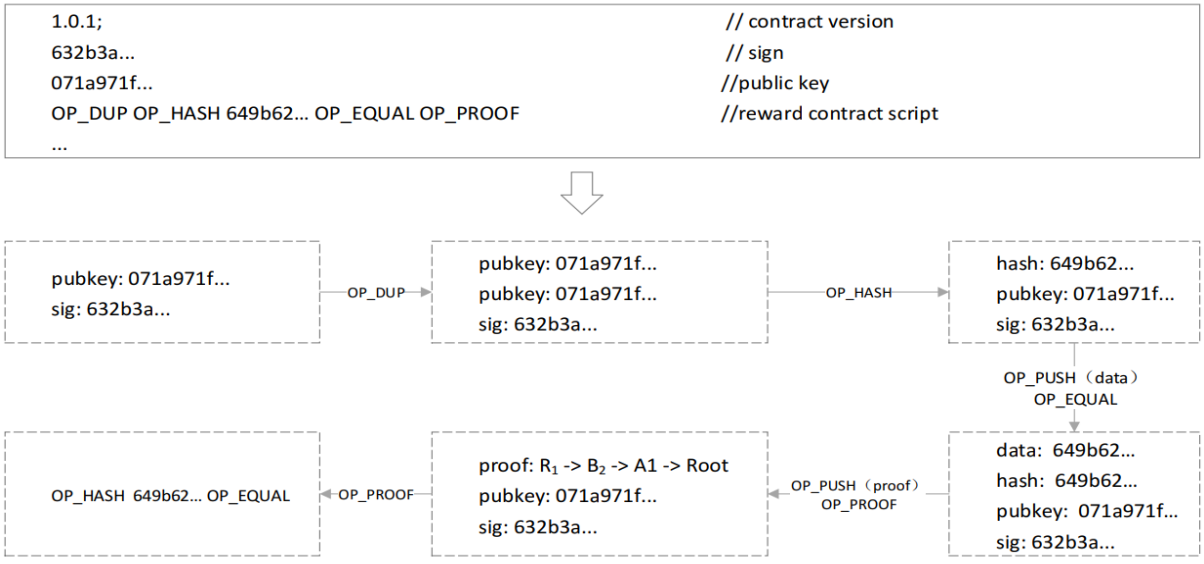


Figure 9 Non-Turing-complete smart contract

그림 9 비튜링완전 스마트 계약

실행은 위에 설명되어 있습니다. 증거를 올바르게 제공하는 저장자가 보상을 얻습니다. 그러나 공개 키 (OP_HASH 에 의해 계산되고 649b62 ...를 얻음)가 필요합니다. 1 단계

스마트 계약은 단순히 주소를 기반으로 한 UTXO 모델을 바탕으로 하며 트랜잭션을 주소와 병합하려면 상위 수준의 샤더 계정이 필요합니다.

스마트 계약은 샤더 자유 시장의 중요한 구성 요소입니다. 이는 저장장부(Store-Book)와 같은 거래 주문의 상태, 변경 및 지불을 보호합니다. 더욱이, 미래에는 다양한 전체 네트워크 데이터장부들(Data-Books)이 스마트 계약에 의해 운영되고 제한될 것입니다. 감시자도 그 중에 포함될 것입니다. 2 단계에서는 계정 기반 튜링완전 스마트 계약(account-based Turing-complete Smart contract)을 설계하고 구현할 것입니다.

4.12 클라이언트

GUI 및 CLI 클라이언트를 사용할 수 있습니다.

저장자 클라이언트 : 샤더 네트워크에 로컬 저장 공간만을 제공합니다. 저장자 클라이언트는 모든 노드 정보를 저장하는 것이 아니라 데이터 파일 및 기여도 확인 파일만 저장합니다. 저장자 클라이언트를 구동하는 노드는 샤더 네트워크의 저장자로만 작동합니다. 로컬 디스크 공간 외에도 가능한 경우 개인 웹 디스크 및 클라우드 저장소와 같은 널리 사용되는 저장소 옵션들도 추가됩니다.

풀 노드 클라이언트 : 샤더 네트워크에 저장 공간뿐만 아니라 번들 블록 생성 및 감시자의 모든 기능을 제공합니다.

감시자 클라이언트 : 풀 노드가 감시자 기능을 열어 감시자 역할을 할 수 있습니다. 감시자는 샤더 네트워크의 변경 상태를 모니터링하고 보안 및 가용성 문제를 해결하기 위한 지침을 보내야 합니다.

증명자 클라이언트 : 증명자는 인가된 온체인 데이터에 대한 인증 서비스를 검사하고 제공할 수 있습니다. 증명자가 독립적인 정보 시스템을 가지고 있으면 이를 API 와 통합할 수 있습니다.

우리는 또한 다방면에 걸친 SDK 와 API 를 제공할 것입니다. 모든 클라이언트들은 완벽한 지갑 기능을 갖습니다. 모바일 라이트 클라이언트도 사용할 수 있지만 작동하려면 풀 노드에 연결되어 있어야 합니다.

4.13 멀티 체인 생태계(Multi-chain Ecosystem)

샤더 프로토콜을 적용하는 샤더 풀들은 함께 멀티 체인 생태계를 형성합니다.

트랜잭션 교환 : 각 Tx 번들에는 샤더 풀에서 정의된 트랜잭션 유형이 포함됩니다. 결국 Tx 번들은 샤더 블록에 포함되어 샤더 네트워크로 전파됩니다.

가치 교환 : 각 샤더 풀은 자체 토큰을 정의할 수 있습니다. SS 는 샤더 풀에서 여러 유형의 트랜잭션을 완료하는데 사용될 수 있습니다. 우리는 샤더 체인과 콘치 체인(Conch Chain) 간의 멀티 체인 아키텍처를 테스트하는 것부터 시작할 것입니다. 이후에는 우리는 정보와 가치의 크로스-체인 거래를 구현하기를 희망합니다. 우리는 동형 채널((homomorphic channel)을 기반으로 하는 라이트닝 네트워크(lightning network) 및 스마트 계약이 샤더 네트워크와 다른 블록체인 네트워크들을 연결시키는데에 훌륭한 방법이 될 것이라고 믿습니다.

4.14 원 페어(One Fair)

원 페어는 다양한 네트워크 장부, 스마트 계약 및 거래 당사자로 구성된 자유 P2P 시장입니다. 샤더 네트워크의 자유 시장은 고빈도(high-frequent) 거래가 나타나는 시장이 아닙니다. 현재 아키텍처는 전체 네트워크 원장 상태 업데이트 및 동기화를 포함하므로 초고빈도(ultra-high frequency) 거래는 실현 가능하지 않습니다. 샤더 시장이 향후 고빈도 거래 또는 그 이상의 초고빈도 거래까지를 처리해야하는 경우 우리는 현금 흐름에서 정보 흐름을 분리하고, 트랜잭션을 다운링크(downlink)로 이동하고, 현금 결제를 위해 업링크(uplink)를 보유함으로써 아키텍처를 조정할 것입니다.

전체 네트워크 원장 : 현재 저장장부(Store-Book), 보상장부(Reward-Book), 증명장부 (Proof-Book)를 포함합니다.

거래 당사자들 : 거래는 스마트 계약에 따라 집행될 수 있습니다. 일단 네트워크에 전파되면 노드가 오프라인일 때도 거래가 수행됩니다.

스마트 계약 : 중앙화된 시장 조성자가 없기 때문에 샤더 네트워크는 매치메이킹 (matchmaking)이 일어나는 때에 의존합니다. 현재 구매자의 거래들의 순서는 매치메이킹이 일어날 때 고려되지 않고 대신 가격과 요구되는 저장 용량이 매치메이킹을 결정합니다.

자유 가격 결정(Free Pricing) : 이상적으로 거래는 거래 당사자들이 가격을 책정하고, 시장이 매칭하며, 스마트 계약에 의해 실행됩니다. 그러나 초기 단계에서 샤더 체인은 거래 당사자들의 가격 결정 및 거래 프로세스를 용이하게 하기 위해 시장 참조 가격 및

매치메이킹을 제공하는 가격 수집가(price collector) 및 통보자(notifier) 기능을 수행합니다.

수수료(Commission) : 수수료는 스마트 계약을 실행하는데 필요한 토큰입니다. 수수료는 시장 조성자가 거래를 할 때 금액이 결정되고 부과됩니다. 모든 수수료는 샤더 토큰 SS 로 정산됩니다. 미래에 수수료는 면제될 수 있습니다. 예를 들어, 시장 매칭을 돕거나 거래를 컨펌하는 사람들의 경우 수수료가 면제가 될 수 있습니다.

샤더 체인 : 샤더 체인은 항상 저장 공간의 판매자가 되며 항상 충분하고 안정적인 저장 공간 공급을 보장합니다.

4.15 인증 메커니즘(Authorization Mechanism)

어느 정도의 감독 및 감사는 여러 상황에서 여전히 필요하며 양 당사자의 인식하에 이루어져야 합니다. 이 아이디어를 바탕으로 샤더 프로토콜은 계정의 권한 부여 및 감사를 돕는 기본 프레임워크를 제공합니다. 이는 개인, 기업 및 규제 기관이 사용자를 대신하여 데이터에 액세스하는 것을 수월하게 합니다.

계정 인증(Account Authorization) : 먼저 분류 및 감사는 데이터 소유자의 인식 및 승인을 받아야 합니다. 모든 데이터 소유자에게 신용을 부여하기 위해 BIP39 [9]의 멀티레벨 (multilevel) 지갑 시스템을 모방합니다. 우리는 BIP44 [10]의 경로 정의 모드(path definition mode)와 이더리움 EIP85 [11]의 논의를 참고할 것입니다. 아래에 경로를 소개합니다 :

m/purpose'/coin_type'/ account' /change/address_index

감사(Audit) : 블록 및 거래 정보가 공개 되더라도 거래 당사자가 익명으로되어 있기 때문에 정보를 감사하기가 어렵습니다. 감사자는 계정 정보의 잠금을 해제하고 교차 확인하기 위해 두 거래 당사자의 승인이 필요하며 거래 정보를 감사하기 위해 이 절차를 반복해야 합니다. 감사 결과는 체인에 기록됩니다(on-chain). 일괄 인증(Batch authorization)은 상용 어플리케이션을 위해 반복적인 절차를 피하는 목적으로 설계될 수 있습니다. 특히 C 터미널 사용자가 B 터미널 사용자를 신뢰할 때 더욱 그렇습니다.

KYC : 샤더 프로토콜은 사용자를 식별하지 않습니다. 고급 사용자는 스스로 사용자 식별을 구현하는 방법을 결정할 수 있습니다.

정보 분류 : 정보 분류는 액세스 제어와 마찬가지로 서로 다른 파생 키들(다른 HD 경로)로 서로 다른 데이터 개체들을 암호화합니다. 파생 키는 데이터의 특정 부분만 해독할 수 있습니다. 키 생성 및 데이터 암호화를 위한 복잡한 논리가 필요합니다.

4.16 악의적인 공격

51% 공격 : 샤더 프로토콜을 포함한 모든 블록체인 시스템을 괴롭히는 문제이며 이를 피할 방법이 없습니다. 샤더 프로토콜은 51 % 공격 확률을 줄이기 위해 PoS 및 DPoS 를 사용하여 블록을 생성하고 PoC 를 추가하여 적합한 블록 생성자를 선택합니다.

시빌 공격 : 샤더 네트워크는 적어도 3 개의 인접한 노드를 확인하기 위해 하나의 트랜잭션이 필요합니다. 이것은 공격자가 대상 노드 주변의 위상(topology)를 계산하고 인접 노드 중 하나로 위장할 수 있는 경우가 아니면 시빌 공격의 가능성을 크게 줄입니다. 더 많은 노드가 네트워크에 참여할수록 시빌 공격의 확률이 낮아집니다. 초기 단계에서 시빌 공격을 피하기 위해 공식 노드의 주소가 공개적으로 나열되어 클라이언트에 포함됩니다. 3 개의 체크 노드에 하나의 공식 노드가 있는 한 시빌 공격은 피할 수 있습니다.

데이터 기만(Data Deception) : 저장자가 인접 저장자에게서 데이터를 얻어 무작위 검사 요청시 매우 짧은 시간 내에 올바른 머클 라우트(Merkle route)를 작동하면, 검사를 통과하여 데이터를 저장하지 않더라도 보상을 얻습니다. PoST 는 그 가능성을 줄이기 위해 채택되었습니다. 속이는 노드는 신용이 깎이거나 샤더 네트워크에서 영구적으로 추방됨으로써 불이익을 받게 됩니다.

데이터 인질극(Data Hijacking) : 저장자가 마지막 데이터 샤드를 제공하기를 거부하여 데이터 빈 개체의 조합(assembly)을 방해하고, 샤드를 인질로 잡아 대가를 요구하는 것. 샤더 프로토콜에서는 데이터가 분할되어 여러 개의 복제본으로 함께 저장됩니다. 저장자는 어느 샤드가 마지막 샤드인지를 반드시 알 필요는 없습니다. 샤드가 있는 모든 저장자가 악의적인 공격자에 의해 손상되지 않는 한 마지막 샤드는 다른 저장자에게서 얻을 수 있습니다. 이러한 공격은 발생할 확률이 희박합니다. 샤더 네트워크가 훨씬 더 분리(discrete)될수록 데이터 인질극이 발생할 확률은 더욱 낮아질 것입니다.

데이터 삭제(Data Erasure) : 저장자가 저장 인센티브가 너무 낮다고 생각하거나 단순히 데이터를 계속 저장하지 않으려고 하여 데이터를 삭제하는 것을 말합니다. 다중 복제는 데이터 손실을 방지하기 위해 도입되었습니다. PoST 전략은 보상의 상당 부분이 저장 기간의 끝에서 지불되도록 조정할 수 있습니다. 가장 효과적인 방법은 노드의 신용을 낮추는 페널티를 적용하여 저장자가 장래에 받을 수 있는 보상을 줄이는 것입니다.

4.17 비전

4.17.1 데이터 가용성

CAP 이론에 따르면 스토리지 시스템이 일관성, 가용성 및 분할 내성(partition tolerance)과 같은 세 가지 보장 중 세 가지 모두를 동시에 제공하는 것은 불가능합니다. 구체적으로 N = 복제본 수량, W = 쓰기 조작에 필요한 쓰기 복제본 수량, R = 쓰기 조작에 필요한 읽기 복제본 수량이라고 해봅시다. 전략은 가치를 NWR 에 할당하고 CAP 조합을 얻는 것입니다. 예를 들어, Amazon 은 N3W2R2 를 채택합니다. 즉, 두 개의 데이터 복제본을 얻을 수 없는 경우 영향을 받는 데이터는 읽기만 가능하며 더 이상 쓰기는 할 수 없습니다. 우리는 계속 연구하고 선도적인 클라우드 스토리지 공급자(Amazon, Facebook, Aliyun)를 참할 것이며, 또한 우리의 데이터 가용성을 최적화할 것입니다.

기존의 RS 이레이저 코드(RS erasure code)와 더불어 데이터 이레이저(data erasure)에서 컴퓨팅 전력 및 네트워크 I / O 소비를 줄이기 위해 FaceBook 및 캘리포니아대학에서 제안한 XORing Elephants 와 같은 SIMD acceleration 및 LRC (Locally Repaureable Codes) 이레이저 알고리즘을 고려하고 있습니다.

4.17.2 디지털 자산 관리

많은 부동산 판매 센터에는 스마트 기기의 도움을 통해 은행 계좌 개설과 일부 예금 동결을 할 수 있는데, 이를 통해 당신의 자산을 증명할뿐만 아니라 이행담보보증금 혹은 계약금(earnest money)으로 사용할 수 있습니다. 이 돈은 실제로 부동산 판매자에게 지불하지 않고 구매자의 은행 계좌에 남아 있지만, 판매자는 이 돈을 통해 구매자가 구매 의향이 있음을 알 수 있습니다. 그러나 비은행(non-banking) 디지털 자산을 효과적으로 증명할 수 있는 방법은 없습니다. 예를 들어, 거래 계정에 있는 토큰의 소유권을 증명하는 것은 어렵습니다. 샤더 프로토콜은 증명자 및 블록체인의 추적 가능하고 변경 불가능한 속성을 기반으로 하는 증거 체인(evidence chain) 통해 신뢰할 수있는 자산증명(PoA, Proof of Assets)을 샤더 네트워크에 제공합니다. 우리는 디지털 자산 관리 및 인증을 위한 완벽한 솔루션을 제시할 것입니다.

스마트 계약을 통해 신뢰할만한 디지털 자산은 낮은 신뢰 혹은 신뢰가 없는 환경에서도 자동으로 거래될 수 있습니다. 자산 관리 및 인증을 위한 다양한 방법들이 만들어 질 수 있습니다. 예를 들어, 스마트 계약에 정의된 시간 또는 조건이 되었을 때 공용

주소(public address)로 자산을 자동으로 보낼 수 있습니다(디지털 자산 주소의 소유권을 증명하는 zero-knowledge proof 와 유사).

4.17.3 샤더 파일 시스템

샤더 파일 시스템(SFS, Sharder File System)은 CloudAqua 위에 업그레이드되어 단일 노드 및 데이터베이스의 트래픽을 늘리고 동시에 다중 프로세스 읽기를 허용합니다. 또한 여러 조각 파일의 IO 성능을 향상시킵니다. SFS 는 Ext4, HFS + 및 NTFS 와 같은 일반적인 로그 파일 시스템과 호환되므로 다양한 운영 체제 또는 물리적 환경의 노드들에 배포하기가 더 쉽습니다.

4.17.4 인공 지능

지난 수년간의 하드웨어 개발과 마찬가지로 AI 는 지도 학습(supervised learning), 대립 네트워크(antagonism network) 등에서 크게 향상되었습니다. 블록체인은 무수히 많은 개방된 데이터가 있는 분야이며 샤더 프로토콜은 분산 스토리지 네트워크를 구축합니다. 태그를 지정하고, 데이터를 분류하고, AI 를 교육하는 방법을 연구하는 것은 흥미로운 일입니다. 지속적인 인공 지능 학습은 샤더 네트워크가 보다 지능적이고 안전하며 효율적이 되도록 도와줍니다. 단기적으로 AI 교육이 보안 전략을 개선하고 감시자가 샤더 네트워크를 보다 더 잘 감시 및 조정할 수 있게 해줄 것으로 기대합니다. 또한, 잠재적인 악성 공격을 예측하고 개입할 수 있습니다. 이를 통해 샤더는 자율적이고 스마트한 네트워크가 될 것입니다.

5. 샤더 체인 (Sharder Chain)

샤더 프로토콜을 적용한 최초의 상용 퍼블릭 체인이고, 0 번째 샤더 풀(Sharder-Pool0)이며, 샤더 네트워크의 초석(cornerstone)입니다. 샤더 프로토콜의 모든 기능은 샤더 체인에서 테스트 되고 적용됩니다. 한편, 샤더 체인은 상용 퍼블릭 체인으로서 사용자 친화적인 계정 모델, 디지털 자산, 보증된 트랜잭션, 사용자 정의된 API(customized API) 및 운영 지원 시스템과 같은 몇 가지 특징을 가지고 있습니다.

5.1 노드와 네트워크

샤더 체인은 업그레이드 된 Kad 프로토콜을 사용하여 P2P 네트워크를 구성합니다. 충분한 노드가 있는 안정적인 네트워크를 신속하게 구축하기 위해 샤더는 저전력 소모 마이크로 노드 마이너(Sharder Hub) 및 올인원 스토리지-마이닝 마이너(Sharder Box)를 출시할 예정입니다.

샤더 허브(Sharder Hub)는 유휴 디스크 용량을 샤더 네트워크에 편리하게 연결할 수 있을뿐 아니라보다 안정적인 온라인 시간을 보장합니다. 임베디드 클라이언트를 통해 구성 작업이 필요없는(configuration-free) 샤더 허브가 스토리지를 즉시 채굴 및 공유 할 수 있게 합니다. 샤더 박스(Sharder Box)는 감시자로 일하면서 저장 공간 공유 및 블록 생성을 하여 여러 가지 보상을 얻을 수 있습니다.

5.2 기능 모델(Funtion Model)

그림 10 샤더 체인 기능 구조

블록체인 레이어(Blockchain Layer) : P2P 네트워크, UTXO 모델, 분산 원장, 글로벌 장부, Sharder 토큰을 포함한 필수 블록체인 모듈들로 구성됩니다.

데이터 레이어(Data Layer) : 샤더 프로토콜에 정의된대로 데이터 운영, 데이터 샤딩, 복제 및 감시자와 증명자의 역할을 구현합니다.

자산 레이어(Asset Layer) : 사용자 친화적인 샤더 계정 모델을 구축하고 토큰 및 데이터 개체를 계정과 연결하여 디지털 자산 모델을 형성하고 디지털 자산 관리를 제공합니다.

모듈 레이어(Module Layer) : 기본 모듈을 추상화(abstracts) 및 패키지화하고 스마트 계약을 기반으로 한 다양한 트랜잭션 모델을 제공합니다.

인터페이스 레이어(Interface Layer) : 블록체인 및 분산 스토리지 서비스의 사용을 용이하게 합니다.

기여도 정량화 : 샤더 네트워크에 대한 다양한 역할의 기여도를 정량화합니다. 역할에 따라 정량화 수식이 다릅니다. 기여자에게는 보상이 주어지고 악의적인 행동에는 불이익이 주어집니다. 기여는 샤더 계정과 연관되어 있습니다.

운영 지원 : 샤더 네트워크에 대한 기업의 액세스를 용이하게 하고 운영 품질을 향상시키는 통계 및 분석을 기업에 제공합니다.

5.3 샤더 계정

샤더 계정은 더 이상 개별 주소(개인 키로 관리)가 아닙니다. 대신 휴대전화 번호와 이메일 주소가 계정을 식별하는데 사용됩니다. 비트코인의 추적 가능하고 감사 가능한 UTXO 모델을 계속 사용할 것입니다. 계정 모델은 UTXO 모델 위에 구축됩니다.

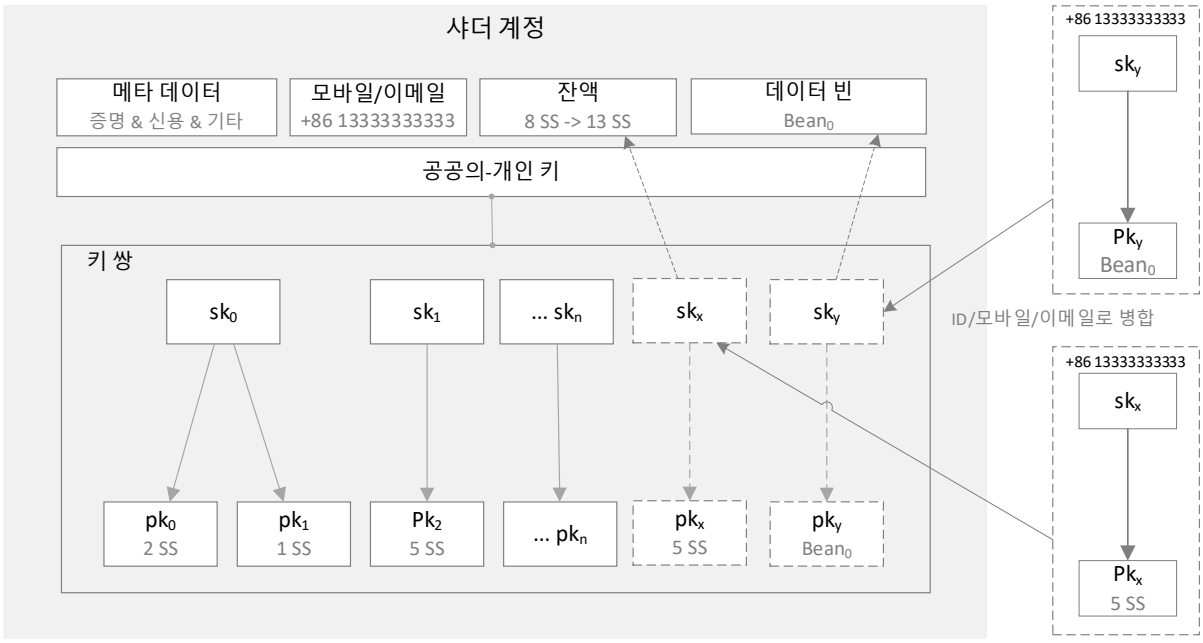


그림 11 샤더 계정

샤더 계정의 공개 키와 비공개 키는 비트코인의 HD 지갑과 유사합니다. 시드(seed)는 샤더 계정의 보안을 책임지고 시드의 개인 키는 사용자가 보유합니다 (암기를 용이하게하기 위해 BIP44 의 서명 세트가 제공됩니다). 샤더 계정 모델은 자동으로 사용자를 식별하고, 디지털 자산을 계정 잔액과 병합합니다(비트코인의 UTXO 모델의 혼란스러운 주소 변경없이). 시드가 안전하게 저장되어있는 한, 계정은 안전합니다. 키는

타원 곡선 알고리즘(elliptic curve algorithm) [13]에 의해 생성되고 서명은 EC-KCDSA [14]에 의해 구현됩니다.

이 계정 모델에는 고성능, 대용량 스토리지 및 안정적인 온라인 상태를 갖춘 풀 노드가 있어야 모든 체인을 검색하고 계정 정보를 얻을 수 있습니다. 감시자는 효율적인 데이터 색인(indexing) 및 캐싱(cashing) 서비스를 제공할 수 있습니다.

5.4 디지털 자산

예상 할 수 있듯이 다양한 물리적 자산이 디지털화됩니다. 샤더 프로토콜은 기업 및 개인에게 분산 데이터 스토리지 및 디지털 자산 관리 서비스를 제공합니다.

샤더 계정의 디지털 자산은 샤더 네트워크에서만 관리하고 교환할 수 있습니다. 디지털 자산을 외부 세계와 교환하기 위해서는 아직 유가증권 거래소(security exchanges)가 필요하고, 디지털 자산 거래소 또는 에이전트 개발이 필요합니다. 샤더 체인은 디지털 자산을 관리하고 교환하기 위해 중앙화 또는 탈중앙화 거래소 및 크로스-체인 시장조성 프로토콜(cross-chain market-making protocols)들과 협력할 것입니다.

5.5 보증 거래(Guaranteed Trade)

DApp 들과 비즈니스의 편리한 거래를 돕기 위해 샤더 체인은 스마트 계약을 바탕으로 보증 거래 모델들을 패키지화 합니다. 여기서 "거래"는 블록체인 시스템에서의 "트랜잭션"과 동일하지 않습니다. 샤더 체인의 보증 거래는 Paypal 또는 Taobao 와 같은 중간 보증인(endorsers) 대신 스마트 계약을 사용합니다.

보증 거래는 판매자 계약서에 거래 자산 (SS, 디지털 자산, 기타 자산 등)을 동결시키는 스마트 계약을 자동으로 생성합니다. 구매자가 동의한 금액의 SS 를 판매자 주소로 지불하면 스마트 계약은 해당 자산을 구매자 주소로 전송합니다. 현재 자동 거래 디지털 자산은 샤더 네트워크 안에 있는 자산들로 제한됩니다. 더 많은 증명자가 네트워크에 연결될수록 보증 거래의 거래 가능한 디지털 자산이 증가할 것입니다.

사전 공인 보증 거래(Pre-authorized Guaranteed Trade) : 양 거래 당사자들은 어느 정도의 보증금을 SS 로 지불합니다. 경매의 경우 모든 계약금 주소와 자산 주소가 스마트 계약에 의해 동결되며, 상태가 전체 네트워크와 동기화됩니다. 스마트 계약은 신용 카드의 사전 승인과 비슷하게 구매자와 판매자 모두의 시간 제한 개인 키 (TPK, time-limited private keys)를 보유합니다.

tpk：시간제한 개인키, t 이후 만료됨
pk：개인 키
t：유효 기간, 단위는 초, 22비트(bit), 최대값은 30일
타임스탬프：tpk의 시작 지점

그림 12 시간 제한 개인 키(TPK)

거래가 완료되면 스마트 계약이 자동으로 TPK 를 사용하여 토큰과 디지털 자산을 전송합니다. 그렇지 않으면 거래가 취소되고 주소가 동결해제(unfrozen) 됩니다. 악의적인 입찰의 경우 예치금이 압수되고 계정의 PoC 가 낮아집니다.

6. 샤더 커뮤니티

비트코인 커뮤니티는 사토시 나카모토가 비트코인에 대한 논문을 발표한 이후 비트코인 운영에 크게 기여했습니다. 우리는 커뮤니티가 없이 비트코인이 있을 수 없다는 것을 알고 있습니다. 따라서 우리는 커뮤니티의 지지와 지혜로 샤더 프로토콜을 지속적으로 개선하기를 희망합니다.

샤더 위원회 : 샤더 재단 구성원, 암화화폐 전문가 및 커뮤니티 구성원으로 구성됩니다. 커뮤니티 운영을 담당하고 토론 및 의견개진을 지지하며 커뮤니티 리저브(reserves)를 관리합니다.

온라인 플랫폼 : 공식 웹 사이트, 회원들이 자유롭게 의견을 말할 수 있는 텔레그램 그룹, 샤더 커뮤니티, 공식 QQ 그룹, 공식 위챗 그룹을 포함합니다.

보상 규칙 : 샤더 위원회는 커뮤니티 기여 보상을 기록, 감사 및 발표합니다. 당분간 보상은 S = N % × M + 50 × N 으로 계산되며, 여기서 S 는 보상 된 SS 수량, N 은 정수 인자 (0 <N ≦ 5), M 은 에어드롭된 주소에 있는 원래 토큰 잔액 (M≥100)을 뜻합니다.

샤더 위원회는 보상 규칙을 지속적으로 개정하고 개선하며 "샤더 커뮤니티 백서"에서 커뮤니티가 한 일과 리저브의 사용에 대한 월간 보고를 제공합니다.

비전 : 샤더 프로토콜의 개발은 샤더 코드의 프로그래밍, 검사 및 테스트에 참여하는 블록체인 애호가의 참여에 달려 있다고 믿습니다. 또한 샤더 체인을 테스트하기 위해 퍼블릭 체인, 기업 및 개인이 필요합니다. 우리는 커뮤니티가 샤더 프로토콜을 홍보하고, 피드백을 제공하며, 우리와 함께 자유롭고 개방된 샤더 커뮤니티를 구축 할 수 있기를 진심으로 바랍니다. "샤더 커뮤니티 백서"에서 운영 및 보상에 대한 자세한 규칙을 확인할 수 있습니다.

7. 어플리케이션

롱테일(Long-tail, 하위 80%의-역자 주) 고객, 고빈도 거래 및 고빈도 사용은 중국 IT 업계의 혁신을 주도하고 개혁을 이끌었습니다. 우리는 더 많은 기업, 사용자 및 네트워크들이 샤더 프로토콜을 적용해야만 장기적인 개발이 가능하다고 굳게 믿습니다.

블록체인의 탈중앙화되고 불변하며 추적 가능하고 영구적인 온라인 상태 속성은 공공 복지, 사물인터넷, 유통망 또는 공유 경제 분야에 큰 이익을 가져다 줄 것이라고 믿습니다. 우리는 다음과 같은 블록체인 기반 비즈니스 애플리케이션들의 연구 및 개발을 위해 파트너사들과 협력하고 있습니다.

7.1 빈 클라우드(Bean Cloud)

P2P 금융, 소액 대출, 소비자 금융, 전자 상거래, ERP 시스템 등을 제공하는 데이터 저장, 인증 및 보안 플랫폼 입니다. 전자 계약, 지불 문서 및 투자 기록과 같은 데이터를 체인에 저장하고, 블록체인의 추적가능하고 조작 불가능한 속성에 기반하여 보안 인증서와 법적 증거를 제공합니다.

7.2 샤더 매트릭스(Sharder Matrix)

유전 정보, 성장 기록, 의료 기록 등 개인적인 생물학적 데이터를 저장하는 어플리케이션 입니다. 미래에는 사고와 기억과 같은 것들도 데이터로 저장할 수 있다고 감히 추측합니다. 데이터가 누적되면 개인 샤더 매트릭스가 형상을 갖춰갈 것입니다.

7.3 샤더 브레인(Sharder Brain)

인공 지능, 스마트 기기, 사물 인터넷 및 비지도 학습(unsupervised learning)의 획기적인 발전 덕분에 샤더 브레인은 개인 및 기업에 데이터 보안, 데이터 배포 조정, 데이터 분석, 데이터 검색, 데이터 경고(데이터 보안 경고, 생체 신호 경고) 등의 스마트 데이터 서비스를 제공할 수 있을 것으로 확신합니다.

7.4 원 페어(One Fair)

샤더 체인 및 샤더 체인을 기반으로 하는 자유 시장은 궁극적으로 투명하고 공개적이고 자유로운 P2P 교환이 이루어지는 개인 데이터 장(원 페어)을 형성합니다. 거래되는 대상은 저장 공간, 디지털 자산, 인증된 데이터, 귀중한 정보 등이 될 것입니다. 예를 들어, 개인은 생체 데이터를 의학 연구 기관에 판매 할 수 있습니다. 유휴 데이터는 현금처럼 가치가 떨어질 것이고 원 페어는 궁극적으로 데이터가 효율적으로 유통되도록 할 것입니다.

8. 개발 계획

8.1 로드맵


2017
Q1
콘치 체인 백서
Q2
콘치 체인 R&D
콘치 커뮤니티
QQ 그룹, 공식 웹사이트, 위챗
Q3
콘치 테스트 네트워크
CPOS 합의 마이닝, 멀티 터미널 클라이언트, 콘치 포인트
빈 클라우드
블록체인 인증 및 공증
Q4
샤더 백서
크로스체인 스토리지 프로토콜, 멀티 체인 아키텍쳐, 신용 모드
샤더 체인 R&D
깃헙에 소스 오픈
빈 클라우드 알파
10개 e-커머스 및 핀테크 플랫폼들이 내부 테스에 참여함
2018
Q1
샤더 커뮤니티
중국어 및 영어 커뮤니티, 로컬 파트너, 보상에 기반한 과업, 커뮤니티 에어드랍
샤더 체인
샤더 프로토콜의 적용, 데이터 샤딩, 풀 노드 클라이언트 터미널
Q2
빈 클라우드 베타
100개 비즈니스 연결됨, 본 클라우드(Born Cloud) 연결됨
감시자와 증명자
데이터 분배 조정, 데이터 색인 서비스, 신용 모델
Q3
샤더 허브
WRTnode 마이크로 마이너, 스토리지 클라이언트 터미널
라이트 클라이언트
모바일 지갑, 웹/라이트 지갑, 토큰 거래소
스마트 계약
튜링완전 스마트 계약, 보장된 거래 모델, 가상 머신
Q4
멀티 체인 마이닝
콘치 네트워크, 퀀텀 블록체인 및 프라이빗 클라우드에 샤더 프로토콜 적용
...
2019

그림 13 샤더 로드맵

8.2 수익 모델

샤더 재단은 오픈 소스 샤더 프로토콜을 비영리 방식으로 운영합니다. 그러나 우리는 연구 개발 및 운영에 지속적으로 투자하기 위해 샤더 체인을 기반으로 한 폐쇄 소스 및 프리미엄 상업용 어플리케이션들 개발하고 다음과 같은 방법으로 수익을 올릴 것입니다.

수익 모델	설명
토큰 가치 상승	샤더 생태계가 발전할수록 내재적 가치는 토큰 가격의 상승으로 나타날 것입니다.
스토리지 거래 수수료	스토리지 거래량 및 매출액이 특정 수준에 도달하면 사용자들에게 수수료 명목으로 샤더 토큰을 청구하기 시작할 것입니다. 이 수수료는 샤더 네트워크를 유지 관리하는데 사용됩니다.
Dapp 서비스 비용 빈 클라우드, 샤더 매트릭스, 샤더 브레인, 원 페어	빈 클라우드는 연간 서비스 수수료를 기업에 부과할 것이며 이는 기업들이 늘어나면 상당한 수입원이 될 것입니다.
기술 서비스 비용	Dapp 들에게 스마트 계약 및 거래 모델과 같은 맞춤형 기술 서비스에 대한 비즈니스 수수료를 청구합니다.
광고비	샤더 체인은 B 터미널 및 C 터미널 사용자가 어느 정도 축적되면 광고를 게시하고 사용료를 부과할 것입니다.

9. 감사의 말

이 백서의 구성 과정에서 Xia Zhang, Xinrong Zuo, Aiping 의 감수 조언에 감사드립니다. Fan Wang 이 Reed-Solomon 알고리즘에 대해 연구하고 테스트한 시간과 노력에 감사드립니다. 이 백서는 분산 웹 시스템 IPFS [15]와 분산 클라우드 스토리지 Storj [16]의 설계 및 Github 의 코드를 참조하고 배웠습니다.

참고문헌

[1] I. Baumgart, S. Mies. S/kademlia: A practicable approach towards secure key-based routing, (2007). http://www.tm.uka.de/doc/SKademlia 2007.pdf.
[2] Wiki. Erasure Code. https://en.wikipedia.org/wiki/Erasure_code
[3] James S. Plank*. A tutorial on reed-solomon coding for fault-tolerance in raid-like systems, (1996). http://web.eecs.utk.edu/~plank/plank/papers/CS-96-332.pdf.
[4] James S. Plank. Tutorial on Erasure Coding for Storage Applications, (2013)http://web.eecs.utk.edu/~plank/plank/papers/2013-02-11-FAST-Tutorial.pdf
[5] Wiki. Reed–Solomon Error Correction. https://en.wikipedia.org/wiki/Reed–Solomon_error_correction
[6] R.C. Merkle. Protocols for public key cryptosystems, (April 1980). http://www.merkle.com/papers/Protocols.pdf
[7] Zcash Blog. Explaining SNARKs. https://z.cash/blog/snark-explain.html
[8] CPOS. Conch Chain. http://www.conchchain.org/
[9] Bitcoin. bip-0039. https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki
[10] Bitcoin. bip-0044. https://github.com/bitcoin/bips/blob/master/bip-0044.mediawiki
[11] Ethereum. Eips. Standardizing HD wallet paths for Ethereum Standard Tokens. https://github.com/ethereum/EIPs/issues/85
[12] University of Southern California & Facebook. XORing Elephants: Novel Erasure Codes for Big Data. https://arxiv.org/pdf/1301.3791.pdf
[13] Yung, M., Dodis, Y., Kiayias, A., Malkin, T., & Bernstein, D. J. (2006). Curve25519: New Diffie-Hellman Speed Records. In , Public Key Cryptography - PKC 2006 (p. 207).
[14] KCDSA Task Force Team. The Korean Certicate-based Digital Signature Algorithm. http://grouper.ieee.org/groups/1363/P1363a/contributions/kcdsa1363.pdf
[15] IPFS. https://ipfs.io/
[16] Storj. https://storj.io

부록

부록 A 네트워크 운영의 정의

1. PING – 노드의 온라인 여부 검사
2. STORE – KVP 를 DHT 에 저장
3. FIND NODE – DHT 에서 버킷의 요청 키-값(key-value)에서 가장 가까운 K 노드를 반환

4. FIND VALUE - DHT 에서 키-값을 반환

부록 B 데이터 운영의 정의

1. PUT – 데이터를 저장
2. GET – 데이터 취득
3. WATCH – 데이터 검사 및 조정
3.1 SETUP – 체크 코드 생성을 위한 초기 구성 설정
3.2 PROVE – 증명 생성
3.3 VERIFY – 증명 검증
3.4 REPAIR – 데이터 배포 조정

부록 C 트랜잭션 운영의 정의

1. ADD ORDER – 트랜잭션 오더 생성
2. MATCH ORDER – 트랜잭션 오더 매치
3. PROC ORDER – 트랜잭션 오더 처리
4. REPAIR ORDER – 트랜잭션 오더 보수
5. DROP ORDER – 트랜잭션 오더 포기