

프롬(Prom)

해당 번역본은 이용자가 디지털자산 관련 정보를 편리하게 확인할 수 있도록
단순 참고용으로 번역된 것이며, 투자 권유를 목적으로 하지 않습니다.
그 내용이 정확하지 않을 수 있으므로 정확한 정보 습득을 위해서는
원문을 참고하시거나 원문 작성 주체 측에 문의하시기 바랍니다.
두나무는 해당 번역본 내용의 진실성 및 정확성을 보장하지 않고,
제공된 정보에 의한 투자 결과에 대하여 어떠한 책임도 지지 않으며,
투자 손실은 투자자에게 귀속됩니다.

시작하기(Welcome)

프롬의 공식 독스(docs)에 온 것을 환영한다. 프롬은 폴리곤(Polygon) CDK를 사용하여 개발된 최첨단 영지식(Zero-Knowledge, ZK) 블록체인으로, 강력하고 확장 가능하며 효율적인 플랫폼으로 주목받고 있다.

네트워크의 중심에는 프롬의 네이티브 체인 통화인 **PROM**이 있으며, 이는 폴리곤 메인넷의 MATIC과 유사한 역할을 한다. PROM은 트랜잭션, 네트워크 보안, 그리고 다양한 필수 블록체인 기능을 지원하는 방식으로 생태계에서 핵심적인 역할을 한다.

이 문서는 프롬 블록체인을 이해하고 탐색하기 위한 종합 가이드이다. 기술적 세부사항부터 실용적인 가이드까지, 플랫폼의 기능과 통합 방법 등 알아야 할 모든 내용을 다룬다.

도입(Introduction)

프롬 입문 가이드

프롬은 폴리곤 CDK를 기반으로 한 고성능 확장형 네트워크로, 영지식 증명(Zero-Knowledge Proofs, ZKP)의 힘을 활용한다. 이러한 암호학적 증명은 오프체인에서 수행되는 트랜잭션의 진위성과 신속한 완료를 보장하며, 이 과정을 영지식 롤업(ZK-Rollup)이라고 한다.

이 영지식 롤업 시스템은 스마트 계약을 투명하게 실행하고, 검증을 위해 영지식 증명을 생성 및 게시하면서도 이더리움 가상 머신(EVM)과 연산 명령어(opcode) 수준에서 완전한 호환성을 유지한다. 본 독스는 다음과 같은 내용을 심층적으로 다룬다:

- 프롬의 구조와 원리
- 혁신적인 상태 머신(State Machines) 설계 전략
- 메인넷 참여 방법

또한 문서는 프롬 네트워크의 정교한 구성 요소들을 탐구한다. 여기에는 영지식 프루버(zkProver), 영지식 프루버 내 상태 머신의 작동 방식, 그리고 프롬이 목표를 달성할 수 있도록 하는 특수 툴킷이 포함된다.

zkEVM을 통한 이더리움의 확장

분산 원장 기술(DLT)의 삼중 딜레마(trilemma)에 의해, 이더리움은 탈중앙화와 보안을 희생하지 않고는 확장성이 제한된다. 프롬은 이 한계를 극복하기 위해 설계되었다.

프롬은 모든 EVM 연산 명령어를 복제함으로써 EVM을 모방하여, 기존 이더리움 스마트 컨트랙트가 원활히 통합되도록 한다. 프롬의 영지식 롤업은 이더리움 메인넷의 확장성과 트랜잭션 처리 속도를 획기적으로 향상시킨다.

오프체인 연산의 정확성을 검증하기 위해, 프롬은 검증 가능한 영지식 증명을 사용한다. 이러한 증명은 복잡한 다항식 계산에 기반하지만, 오프체인 트랜잭션에 대해 효율적이고 간단한 검증 및 최종성을 제공한다.

이 혁신적인 설계 모델을 채택한 것은 사용자와 개발자 모두의 진입 장벽을 낮추기 위한 전략적 선택이었다. 이는 모든 EVM 연산 명령어를 재구현하여 기존 이더리움 스마트 컨트랙트를 투명하게 통합하는 것을 포함하며, 이는 우리가 새롭게 개발한 기술과 도구들에 의해 가능해졌다. 이에 대한 자세한 내용은 문서 후반부에서 다룰 것이다.

프롬의 주요 이점

- EVM과 동등한 호환성
- 이더리움에서 파생된 보안성
- 영지식 증명으로 강화된 확장성

개발자는 기존 스마트 컨트랙트를 프롬에 손쉽게 배포할 수 있으며, 사용자는 이더리움에서 자산을 이동시켜 오프체인 운영을 수행할 수 있다. 이러한 운영은 그룹화되어 영지식 증명으로 검증되며, 각 트랜잭션의 유효성을 보장하고 오퍼레이터에 의한 사용자 자금 오용을 방지하여 이더리움의 신뢰할 수 있는 보안 프레임워크를 유지한다.

프롬 네트워크는 호환성과 확장성을 효과적으로 조화시켜 어떠한 타협도 하지 않는다.

퀵스타트(Quickstart)

프롬은 EVM과 완전히 동등한 최초의 영지식 확장 솔루션으로, 기존의 모든 스마트 컨트랙트, 개발자 도구, 지갑이 문제없이 작동하도록 보장한다. 프롬은 영지식 증명을 활용하여 트랜잭션 비용을 크게 줄이고 처리량을 대폭 향상시키면서도 강력한 보안을 유지한다.

프롬에서 디앱(dApp)을 개발하는 과정은 이더리움에서의 과정과 동일하다. 단순히 연결을 프롬 RPC로 전환하기만 하면, 훨씬 높은 처리량과 낮은 수수료를 제공하는 환경에서 애플리케이션을 개발할 수 있다. 프롬은 개발자와 사용자 모두에게 매끄러운 EVM 유사 경험을 제공한다. 즉, 블록체인을 개발하거나 상호작용하기 위해 특별한 도구나 새로운 지갑이 필요하지 않다.

개발자는 기존 컨트랙트를 손쉽게 프롬에 배포할 수 있으며, 사용자는 이더리움에서 자산을 전송하여 오픈체인 트랜잭션을 수행할 수 있다. 이러한 트랜잭션은 배치(batch) 처리되며, 각각은 영지식 증명으로 검증되어 정당성이 보장된다.

프롬에 연결하기(Connecting to Prom)

프롬 메인넷에 연결한다.

지갑에서 **네트워크 추가(Add Network)** 입력으로 이동한 후, 아래 표에 제공된 네트워크 세부 정보를 입력하여 프롬 네트워크를 추가한다.

네트워크	RPC URL	체인ID	블록 익스플로러 URL	통화
프롬	https://rpc.prom.io/	227	https://promscan.io/	PROM

프롬 토큰(Prom Token)

PROM은 프롬 생태계를 구동하는 네이티브 토큰으로, 모든 트랜잭션을 지원하고, 사용자에게 거버넌스 권한을 부여하며, 커뮤니티 기여를 장려한다.

1. **가스 수수료 토큰:** 프롬 토큰은 빠르고 원활한 트랜잭션과 컨트랙트 상호작용을 가능하게 한다.
2. **거버넌스:** 프롬 토큰은 사용자에게 의사결정 권한을 부여하여 네트워크의 미래를 형성하도록 한다.
3. **접근성:** 다양한 프롬 네트워크 서비스를 이용하고 참여할 수 있다.
4. **검증:** 자신만의 노드를 설정하여 생태계의 일원이 될 수 있다.

거버넌스(Governance)

블록체인 생태계가 발전함에 따라, 거버넌스 메커니즘도 크게 변화하고 있다. 메인넷 출시 이후 **프롬 다오(Prom DAO)**의 도입은 이러한 진화를 보여주며, 블록체인 영역에서 보다 민주적이고 참여적인 거버넌스 모델로의 전환을 강조한다.

프롬 다오 소개(Introduction to Prom DAO)

프롬 다오는 프롬 네트워크의 거버넌스에서 한 단계 도약한 모델로, \$PROM 토큰 보유자가 네트워크의 미래에 직접적으로 영향을 미칠 수 있도록 역할 기반 구조를 강조한다. 이 구조를 통해 토큰 보유자는 자신의 투표 권한을 다오 내 다른 구성원에게 위임할 수 있으며, 네트워크 의사결정 과정에 참여하고 생태계의 성장으로부터 혜택을 얻을 수 있다.

프롬 다오의 기술적 프레임워크(Technical Framework of Prom DAO)

프롬 다오의 거버넌스 모델은 투명하고 탈중앙화된 기반 위에 구축되어 있다. 의사결정은 특정 소수 집단에 국한되지 않고 네트워크 커뮤니티 전반에 분산된다. 이 모델은 다오의 거버넌스 프로세스에서 지분이자 의사표현 수단으로 기능하는 **\$PROM 토큰**에 크게 의존한다.

의사결정 과정(Decision-Making Process)

프롬 다오의 의사결정은 유연하며, 핵심 컨트랙트 업그레이드, 취약점 대응, 성능 향상, 사용자 경험 개선 등 광범위한 활동을 포괄한다. 이를 통해 네트워크가 사용자와 더 넓은 생태계의 요구에 맞게 지속적으로 적응하고 대응할 수 있도록 한다.

역할 기반 구조(Role-Based Structure)

프롬 다오는 거버넌스 과정을 간소화하기 위해 명확한 역할 지향적 계층 구조를 도입한다:

- **대리인(Delegates):** 커뮤니티의 다른 구성원들로부터 투표권을 위임받은 \$PROM 보유자이다. 이들은 다오 내에서 자신을 지지한 구성원의 이익을 대표하는 중요한 역할을 한다.
- **검증인(Validators):** 제품 개발 과정의 무결성을 유지하고, 제안이 네트워크의 장기 비전과 일치하는지를 보장한다. 초기 투표 단계를 통과한 제안에 대해 추가적인 검토와 검증을 제공한다.
- **멤버(Members):** 다오의 근간이 되는 계층으로, 모든 \$PROM 보유자는 참여 자격을 가진다. 멤버는 제안을 발의하고, 의사결정에 투표하며, 네트워크의 방향성에 기여할 수 있다.

경제적 인센티브(Economic Incentives)

프롬 다오는 적극적인 참여를 장려하기 위해 매력적인 인센티브 모델을 도입한다. 대리인과 검증인은 네트워크 수수료의 **32%**를 공유하며, 이를 통해 네트워크의 경제적 성과와 거버넌스 참여 보상이 직접적으로 연결된다. 이러한 분배는 활발하고 의미 있는 참여를 보상하도록 설계되어, 네트워크의 거버넌스와 발전에 가장 크게 기여하는 사람들이 충분히 보상받을 수 있도록 한다.

토큰 위임 및 보상(Token Delegation and Rewards)

투표권을 위임하고자 하는 토큰 보유자는 **프롬 거버넌스 포털**을 통해 \$PROM 토큰을 예치하고, 선택한 대리인에게 할당되는 투표권을 발행받을 수 있다. 이 시스템은 토큰 보유자와 대리인 간의 공생적 관계를 조성하며, 양측 모두 네트워크의 거버넌스 활동으로부터 혜택을 얻는다.

본격적으로 시작하기(Get started)

이 섹션은 개발자가 고성능 확장 네트워크인 프롬 블록체인에서 작업을 시작하기 위한 필수 가이드를 제공한다.

여기서 지갑을 프롬 네트워크에 연결하는 방법, 새로운 또는 기존 이더리움 스마트 컨트랙트를 배포하는 방법, 그리고 프롬과 이더리움 간 자산을 브리징하는 방법을 확인할 수 있다.

또한 기술 스택, RPC zkNode 설정 방법, 스마트 컨트랙트 배포 절차에 대해서도 배울 수 있다.

프롬은 이더리움과 완전히 호환되므로 개발자와 사용자 모두에게 원활한 경험을 제공한다. 프롬 네트워크를 구축하거나 상호작용하기 위해 특별한 도구나 지갑은 필요하지 않다.

단순히 프롬 RPC로 전환하고, 더 높은 처리량과 더 낮은 수수료를 제공하는 네트워크에서 개발을 시작하면 된다. 이제 프롬 네트워크의 다양한 가능성을 탐험해 보자!

자산을 프롬으로 브리징하기(Bridge Assets to Prom)

프롬은 다용도의 멀티체인 브리지를 제공하여 사용자가 다양한 블록체인 네트워크와 프롬 생태계 간에 자산을 원활하게 전송할 수 있도록 한다. 이 강력한 도구는 상호운용성을 강화하고, 디지털 자산을 관리하는 사용자에게 유연성을 제공한다.

브리지 URL: <https://prom.io/bridge>

지원 체인(Supported Chains)

프롬 브리지는 다음과 같은 여러 블록체인 네트워크를 지원한다:

- 이더리움(Ethereum)
- BSC(바이낸스 스마트 체인)
- 옵티미즘(Optimism)
- 아비트럼(Arbitrum)
- 폴리곤(Polygon)

사용자는 이들 지원 체인 중 어느 쪽에서든 자산을 프롬으로 또는 프롬에서 해당 체인으로 브리징할 수 있으며, 이를 통해 폭넓은 크로스체인 트랜잭션이 가능하다.

자산 호환성(Asset Compatibility)

프롬 브리지의 핵심 기능 중 하나는 광범위한 자산 지원이다. **모든 자산**을 지원 체인에서 프롬으로, 또는 프롬에서 지원 체인으로 브리징할 수 있다. 이러한 포괄적인 호환성 덕분에 사용자는 다양한 토큰과 디지털 자산을 여러 블록체인 생태계 간에 자유롭게 전송할 수 있다.

브리징 절차(Bridging Process)

자산을 브리징하려면 공식 프롬 브리지 인터페이스 <https://prom.io/bridge>를 방문하면 된다. 직관적인 UI를 통해 출발지 체인과 목적지 체인, 그리고 전송할 자산을 쉽게 선택할 수 있다.

브리지 기술(Bridge Technology)

프롬은 두 가지 주요 브리징 기술을 활용한다:

1. **공식 브리지(Official Bridge):** 특정 경로에서 사용되며, 안전하고 신뢰할 수 있는 전송 방법을 제공한다.
2. **하이퍼플레인 기술(Hyperplane Technology):** 특정 경로에서 사용되며, 속도와 효율성을 최적화한다.

어떤 기술이 사용되는지는 사용자가 선택한 브리징 경로에 따라 달라진다.

브리징 소요 시간(Bridging Time)

브리징에 소요되는 시간은 선택한 경로와 네트워크 상태에 따라 달라질 수 있다. 대부분의 트랜잭션은 합리적인 시간 내에 완료되지만, 일부 경우에는 한 시간을 초과할 수도 있다. 따라서 제공된 인터페이스를 통해 트랜잭션 상태를 확인하며 인내심을 갖는 것이 중요하다.

중요한 고려사항(Important Considerations)

- **네트워크 수수료:** 브리징 과정에서 발생할 수 있는 네트워크 수수료를 반드시 확인해야 한다.
- **보안:** 자산의 안전을 보장하기 위해 항상 공식 프롬 브리지 인터페이스를 사용해야 한다.

프롬 브리지를 활용하면 사용자는 여러 블록체인 네트워크 전반에서 유동성을 강화하고 원활한 자산 관리를 경험할 수 있다. 모든 자산을 프롬으로 또는 프롬에서 브리징할 수 있는 기능은 사용자에게 다양한 디지털 자산 포트폴리오를 관리하는 데 있어 비할 데 없는 유연성을 제공한다.

zkNode

zkNode는 프롬 노드를 실행하는 데 필요한 소프트웨어이다. 이는 네트워크 사용자가 프롬 네트워크의 상태를 동기화하고 파악하기 위해 필요한 클라이언트이다. 상태와 최종성에 영향을 미치는 주요 주체는 **신뢰할 수 있는 시퀀서(trusted Sequencer)**와 **신뢰할 수 있는 어그리게이터(trusted Aggregator)**이다.

프롬은 여러 핵심 구성 요소를 통해 이를 달성한다. 아래는 네트워크의 가장 중요한 요소들이다:

- **사용자(Users)**는 RPC 노드(예: MetaMask)를 통해 네트워크에 연결하고 트랜잭션을 **Pool DB**라 불리는 데이터베이스에 제출한다.
- **Pool DB**는 사용자가 제출한 트랜잭션을 저장하는 저장소이다. 트랜잭션은 시퀀서가 이를 배치로 정리할 때까지 풀에 보관된다.
- **시퀀서(Sequencer)**는 Pool DB에서 트랜잭션을 가져와 검증한 후, 유효한 트랜잭션을 배치로 정리하는 역할을 한다. 시퀀서는 모든 배치를 L1에 제출하고 순서를 지정하여, 이 배치들이 L1 상태에 포함되도록 한다.
- **싱크로나이저(Synchronizer)**는 이더맨(Etherman)을 통해 이더리움에서 데이터를 가져와 상태 데이터베이스(State DB)를 업데이트한다.
- **이더맨(Etherman)**은 L1 네트워크 및 스마트 컨트랙트와의 모든 상호작용을 처리하는 저수준 구성 요소이다.
- **상태 데이터베이스(State DB)**는 상태 데이터를 영구 저장하는 데이터베이스(머클 트리는 제외)이다.
- **어그리게이터(Aggregator)**는 시퀀서가 제안한 상태 변경의 무결성을 검증하는 증명을 생성하는 또 다른 노드이다. 이는 영지식 증명이며, 어그리게이터는 이를 위해 **프루버(Prover)**라는 암호학적 구성 요소를 사용한다.
- **프로버(Prover)**는 다수의 배치에 대한 영지식 증명을 생성하고 이를 단일 영지식 증명으로 집계하는 고급 암호학적 도구이다. 이렇게 생성된 유효성 증명은 공개된다.

zkNode의 역할(zkNode roles)

zkNode 소프트웨어는 다양한 역할을 수행할 수 있도록 설계되었으며, 각 역할은 서로 다른 서비스를 필요로 한다. 대부분의 서비스는 별도의 인스턴스로 실행될 수 있지만, **JSON RPC**는 다중 인스턴스 실행이 가능하며, 다른 모든 서비스는 단일 인스턴스에 제한된다.

RPC 엔드포인트(RPC Endpoints)

누구나 RPC 노드 역할에 참여할 수 있다.

필수 서비스 및 구성 요소:

- **JSON RPC**: 별도 인스턴스에서 실행 가능하며, 여러 인스턴스를 지원한다.
- **싱크로나이저(Synchronizer)**: 독립적으로 실행 가능한 단일 인스턴스가 필요하다.
- **엑시큐터(Executor) & 머클트리(Merkletree)**: 별도 인스턴스에서 실행 가능하다.
- **상태 데이터베이스(State DB)**: Postgres SQL을 사용하며 별도 인스턴스에서 실행 가능하다.

싱크로나이저는 반드시 하나만 존재해야 하며, 전용 엑시큐터 인스턴스에 대한 독점 접근 권한을 갖는 것이 권장되지만 필수는 아니다. 싱크로나이저의 역할은 단일 인스턴스로도 효과적으로 작동할 수 있지만, JSON RPC 및 엑시큐터 서비스는 요청량이 많아 성능 저하가 발생할 경우 다중 인스턴스 구성을 통해 성능을 향상시킬 수 있다.

신뢰할 수 있는 시퀀서(Trusted Sequencer)

이 역할은 스마트 컨트랙트에 의해 단일 주체로 제한된다. 신뢰할 수 있는 시퀀서와 관련된 메서드는 특정 개인키 소유자만 실행할 수 있다.

필수 서비스 및 구성 요소:

- **JSON RPC**: 별도 인스턴스에서 실행 가능하며, 여러 인스턴스를 지원할 수 있다.
- **시퀀서 & 싱크로나이저(Sequencer & Synchronizer)**: 반드시 단일 인스턴스에서 함께 실행되어야 한다.
- **엑시큐터 & 머클트리(Executor & Merkletree)**: 별도 인스턴스에서 실행 가능하다.
- **Pool DB**: PostgreSQL을 사용하며 독립적으로 실행될 수 있다.
- **상태 데이터베이스(State DB)**: PostgreSQL을 사용하며 별도 인스턴스에서 실행 가능하다.

JSON RPC는 트랜잭션 수신에 필수적이다. 네트워크 부하에 따라 JSON RPC를 별도 인스턴스, 가능하다면 다중 인스턴스로 실행하는 것이 권장된다. 또한 시퀀서가 실행기에 대한 독점 접근 권한을 갖도록, JSON RPC와 시퀀서가 동일한 엑시큐터 인스턴스를 공유하지 않는 것이 바람직하다.

어그리게이터(Aggregator)

이 역할은 누구나 수행할 수 있다.

필수 서비스 및 구성 요소:

- **싱크로나이저(Synchronizer)**: 독립적으로 실행 가능한 단일 인스턴스이다.
- **엑시큐터 & 머클트리(Executor & Merkletree)**: 별도 인스턴스에서 실행 가능하다.

- **상태 데이터베이스(State DB):** PostgreSQL을 사용하며 별도 인스턴스에서 실행 가능하다.
- **어그리게이터(Aggregator):** 독립적으로 실행 가능한 단일 인스턴스이다.
- **프루버(Prover):** 별도 인스턴스에서 실행 가능한 단일 인스턴스이다.
- **엑시큐터(Executor):** 독립적으로 실행 가능한 단일 인스턴스이다.

프루버는 상당한 하드웨어 요구 사항 때문에 별도 인스턴스에서 실행하는 것이 권장된다. 그러나 필요하다면 다른 모든 구성 요소는 단일 인스턴스로 통합하여 실행할 수도 있다.

앞으로 Synchronizer, State DB, Executor, Prover는 각각 싱크로나이저, 상태 데이터베이스, 엑시큐터, 프루버로 번역한다.

MaticJS

Matic.js는 이더리움의 레이어 2 확장 솔루션인 매틱(Matic) 네트워크와의 상호작용을 용이하게 하기 위해 특별히 설계된 자바스크립트 클라이언트 라이브러리이다. 이 라이브러리는 특히 스마트 컨트랙트 개발 경험이 제한적인 개발자에게 사용자 친화적인 인터페이스를 제공하여 매틱 네트워크의 다양한 구성 요소와 쉽게 상호작용할 수 있도록 한다.

주요 특징(Key Features)

- 매틱 네트워크와의 단순화된 상호작용
- npm 설치를 통한 빠른 설정
- 이더리움과 매틱 간 자산 브리징 지원
- 플라즈마(Plasma) 및 PoS(지분 증명) 브리지 트랜잭션을 위한 특화된 클라이언트 제공

장점(Benefits)

- 1. 개발자 친화적 인터페이스(Developer-Friendly Interface)**
 - **목적:** 블록체인 및 스마트 컨트랙트 개발에 익숙하지 않은 개발자의 기술적 장벽 해소
 - **이점:** 단순화된 학습 곡선을 바탕으로 한 더 빠른 개발 주기
- 2. 자산 브리징 기능(Asset Bridging Capabilities)**
 - **기능:** 이더리움 루트 체인과 매틱 차일드 체인(child chain) 간 자산 이동 지원
 - **이점:** 크로스체인 자산 전송이 필요한 디앱 개발을 용이하게 하여 애플리케이션 기능 강화
- 3. 플라즈마 및 지분 증명 브리지 지원(Plasma and PoS Bridge Support)**
 - **구성 요소:** MaticPlasmaClient와 MaticPoSClient를 통해 각 브리지 트랜잭션 지원
 - **효용성:** 효율적인 트랜잭션 관리를 위한 API 제공, 크로스체인 상호작용의 속도와 신뢰성 향상
- 4. 사기 증명 메커니즘(Fraud Proof Mechanisms)**
 - **특징:** 사기 증명을 사용하여 자산 이동을 수행할 수 있는 메커니즘 내장
 - **보안 강화:** 탈중앙화 애플리케이션에서 중요한 자산 전송의 무결성과 안전성을 보장
- 5. 원활한 통합(Seamless Integration)**
 - **통합 용이성:** 표준 개발 도구 및 환경과의 호환성을 고려하여 설계

- **유연성:** 기존 프로젝트에 큰 재구성 없이 손쉽게 통합 가능

6. 향상된 트랜잭션 효율성(Enhanced Transaction Efficiency)

- **성능:** 매트 네트워크의 높은 처리량과 낮은 트랜잭션 수수료 활용
- **비용 효율성:** 이더리움 메인넷 대비 더 효율적인 트랜잭션 경험 제공

설치, 초기화, ERC20 메서드 및 MaticJS의 기타 기능에 대해 더 알아보려면 [이 링크](#)를 참고할 수 있다. 이 자료는 MaticJS 라이브러리를 개발 프로젝트에서 효과적으로 활용할 수 있도록 상세한 정보와 단계별 가이드를 제공한다.

계정 추상화(Account Abstraction)

사용자는 EOA(Externally Owned Account)를 통해 이더리움과 상호작용하며, 이는 트랜잭션을 시작하거나 스마트 계약을 실행하는 데 필수적이다. 그러나 이러한 방식은 사용자가 플랫폼과 상호작용할 수 있는 방법을 제한하며, 배치 트랜잭션과 같은 과정이 복잡해지고 가스 수수료를 위한 ETH 잔고를 항상 유지해야 한다는 불편함을 초래한다.

계정 추상화는 계정 프로그래밍을 보다 유연하게 만들어 보안성과 사용자 경험을 향상시키는 해결책을 제시한다. 이는 EOA를 스마트 계약에 의해 관리되도록 수정하거나, 스마트 계약이 직접 트랜잭션을 시작할 수 있도록 허용하는 방식을 포함할 수 있다. 이러한 변경을 구현하기 위해서는 이더리움 프로토콜 수정이 필요하다. 또 다른 접근 방식은 현재 프로토콜과 함께 작동하는 보조 트랜잭션 시스템을 도입하는 것이다. 어떤 방법이든, 목표는 스마트 계약 지갑을 통해 이더리움에 접근할 수 있도록 하는 것이다. 이는 기존 프로토콜의 직접적인 기능으로 구현되거나, 추가적인 트랜잭션 네트워크를 통해 실현될 수 있다.

스마트 계약 지갑은 사용자에게 다음과 같은 수많은 이점을 제공한다:

- 맞춤형 보안 조치
- 키 분실 시 계정 복구 옵션
- 신뢰할 수 있는 장치나 사람들 간의 계정 보안 분산
- 다른 사람의 가스 수수료를 대신 지불하거나 대납받을 수 있는 옵션
- 여러 트랜잭션을 단일 동작으로 통합 (예: 스왑 승인과 실행을 결합)
- 디앱 및 지갑 개발자가 혁신적인 사용자 경험을 창출할 수 있는 확장된 가능성

예시 스마트 계약 생성(Creating an example Smart Contract)

OpenZeppelin Wizard & Remix IDE를 사용한 스마트 계약 생성

이 가이드는 OpenZeppelin Wizard를 사용하여 스마트 계약을 자동으로 생성하는 과정을 설명한다. 생성된 스마트 계약은 "Open in Remix" 버튼을 통해 Remix에서 바로 열거나, 원하는 IDE에 복사하여 붙여넣을 수 있다.

초기 단계(Initial Steps)

먼저 웹 브라우저에서 OpenZeppelin Wizard에 접속한다. 화면에는 **Solidity Wizard**와 **Cairo Wizard** 두 가지 옵션이 표시된다.

다음 옵션 중 하나를 선택하여, Solidity(EVM 체인을 위한 이더리움 가상 머신용) 또는 Cairolang(Starknet용)에서 즉시 사용 가능한 스마트 컨트랙트 코드를 생성할 수 있다:

- **ERC20**: ERC-20 토큰 컨트랙트 생성
- **ERC721**: NFT 컨트랙트 생성
- **ERC1155**: ERC-1155 토큰 컨트랙트 생성
- **Governor**: 다오 설정
- **Custom**: 맞춤형 스마트 컨트랙트 작성

NFT 컨트랙트 생성(Creating an NFT Contract)

예시로, 발행(Minting)이 가능한 ERC721 NFT 컨트랙트를 생성하고 라이선스를 지정해 보자.

ERC721 탭 선택(Select the ERC721 Tab)

NFT의 이름(Name)과 심볼(Symbol)을 해당 입력란에 입력한다.

좌측의 체크박스를 사용하여 토큰에 기능을 추가한다:

- **Mintable** 선택
- **Auto Increment Ids** 선택 → 고유한 NFT 발행을 위한 자동 증가 ID

기본 **MIT 라이선스**를 선택하거나 다른 라이선스를 입력한다.

선택한 각 기능에 따라 해당되는 코드가 자동으로 생성된다.

이제 컨트랙트가 준비되었다.

이 과정을 통해 Solidity로 작성된 완전한 NFT 토큰 컨트랙트가 생성된다. 이 코드를 선호하는 IDE에서 사용하거나 Remix에서 열 수 있다.

아래 그림은 자동으로 생성된 NFT 스마트 컨트랙트 코드를 보여준다.

ERC20ERC721ERC1155GovernorCustom

Copy to ClipboardOpen in RemixDownload

SETTINGS

NameSymbol

MyTokenMTK

Base URI

https://...

FEATURES

☒ Mintable

☒ Auto Increment Ids

☐ Burnable

☐ Pausable

☐ Votes

☐ Enumerable

☒ URI Storage

ACCESS CONTROL

☒ Ownable

☐ Roles

☐ Managed

UPGRADEABILITY

☐ Transparent

☐ UUPS

INFO

Security Contact

security@example.com

License

MIT

// SPDX-License-Identifier: MIT

pragma solidity ^0.8.20;

import "@openzeppelin/contracts/token/ERC721/ERC721.sol";

import "@openzeppelin/contracts/token/ERC721/extensions/ERC721URIStorage.sol";

import "@openzeppelin/contracts/access/Ownable.sol";

contract MyToken is ERC721, ERC721URIStorage, Ownable {

uint256 private _nextTokenId;

constructor(address initialOwner)

ERC721("MyToken", "MTK")

Ownable(initialOwner)

{}

function safeMint(address to, string memory uri) public onlyOwner {

uint256 tokenId = _nextTokenId++;

_safeMint(to, tokenId);

_setTokenURI(tokenId, uri);

}

// The following functions are overrides required by Solidity.

function tokenURI(uint256 tokenId)

public

view

override(ERC721, ERC721URIStorage)

returns (string memory)

{

return super.tokenURI(tokenId);

}

function supportsInterface(bytes4 interfaceId)

public

view

override(ERC721, ERC721URIStorage)

returns (bool)

{

return super.supportsInterface(interfaceId);

}

}

컨트랙트 배포(Deploy Contract)

프롬에 연결하기 위해 필요한 데이터와 관련된 최신 업데이트를 반드시 확인하도록 하자.

지갑(Wallets)

프롬이 지원하는 이더리움 지갑은 무엇인가? (What Ethereum Wallets Does Prom Support?)

일반적으로 프롬 네트워크를 사용하면 거래소 및 스마트 컨트랙트를 포함한 모든 이더리움 주소로 안전하게 자금을 전송할 수 있다. 해당 주소의 소유자는 언제든지 이 자금을 접근하고 이를 청구할 수 있다.

프롬 웹 지갑과 호환되는 이더리움 지갑의 최신 목록은 곧 홈페이지에서 확인할 수 있다. 이 지갑들은 프롬과 원활하게 통합되도록 테스트 및 검증이 완료되었다.

더 기술적인 사용자를 위해, 프롬은 이더리움 서명을 통한 주소 제어를 지원한다. 여기에는 네이티브 이더리움 서명 또는 스마트 컨트랙트를 위한 EIP1271 방식이 포함된다.

내 지갑이 지원되지 않거나 메시지 서명을 할 수 없는 경우 어떻게 해야 하는가?(What If My Wallet Is Not Supported or Can't Sign Messages?)

특정 상황에서는 일부 이더리움 주소가 프롬과 직접적으로 사용되지 못할 수 있다:

- 웹 인터페이스에서 아직 지원되지 않은 지갑
- 거래소가 소유한 주소
- EIP1271을 지원하지 않는 스마트 컨트랙트에 연결된 주소

그러나 사용자는 여전히 프롬 계정에서 동일한 주소의 레이어 1로 자금을 인출할 수 있다. 이는 프로토콜이 **자금 소유권은 변경되지 않는다**는 보안 원칙을 유지하기 때문이다. 단, 이 과정은 계정이 프롬 공개키를 한 번도 등록한 적이 없어야 한다.

보안 파트너 (Security Partners)

공식 보안 파트너와 연결하여 프롬 생태계를 위한 독점 혜택을 누리세요.

1. Bailsec 웹사이트, 메일

Bailsec은 블록체인 프로젝트에서 발생할 수 있는 취약점, 보안 문제, 잠재적 위험을 식별하기 위한 종합적인 스마트 컨트랙트 감사를 전문으로 한다. 숙련된 감사인들이 수행하는 이들의 서비스는 스마트 컨트랙트의 보안성과 신뢰성을 강화하여, 창업자가 안전하게 배포할 수 있도록 지원한다. 또한 Bailsec은 고급 침투 테스트 및 KYC 인증 서비스도 제공하며, 블록체인 보안 표준을 개선하려는 그들의 헌신을 보여준다.

버그 바운티 프로그램(Bug Bounty Program)

프롬 버그 바운티 프로그램에 온 것을 환영한다! 프롬은 블록체인 솔루션의 보안성과 신뢰성을 보장하기 위해 최선을 다하고 있다. 이 프로그램은 보안 연구원과 윤리적 해커가 시스템 내 잠재적 취약점을 식별하고 보고하도록 장려하기 위해 마련되었다.

보상(Rewards)

보상은 취약점의 심각도에 따라 분배되며, 간단한 5단계 기준을 따른다:

- **치명적임(Critical):** 최대 \$50,000
- **높음(High):** 최대 \$10,000
- **중간(Medium):** 최대 \$2,000
- **낮음(Low):** 최대 \$500
- **매우 낮음(Informational):** 최대 \$100

최종 보상 금액은 보고된 문제의 잠재적 영향과 악용 가능성에 따라 결정된다.

범위(Scope)

버그 바운티 프로그램은 프롬의 다음 구성 요소를 포함한다:

- 1. 스마트 컨트랙트
- 2. 브리지 메커니즘
- 3. 합의 레이어
- 4. 상태 관리
- 5. 트랜잭션 처리
- 6. 증명 생성 및 검증

중점 영역(Focus Areas)

다음과 같은 취약점에 특히 관심이 있다:

- 사용자 자금의 손실 또는 탈취
- 자금의 영구적 동결
- 트랜잭션 데이터 또는 상태 조작
- 네트워크 무결성 훼손
- 프롬과 이더리움 간 브리지 교란
- 영지식 증명 시스템의 악용
- 합의 실패 또는 네트워크 분열

제출 지침(Submission Guidelines)

버그 리포트는 tech@prometeus.io 이메일로 제출해야 한다. 리포트에는 다음 사항이 반드시 포함되어야 한다:

- 1. 모든 제출물에는 취약점에 대한 명확한 설명과 PoC(Proof-of-Concept, 개념 증명)가 포함되어야 한다.
- 2. 스마트 컨트랙트 취약점의 경우, 작동 가능한 악용(exploit) 코드가 필요하다.
- 3. 취약점을 재현할 수 있는 상세한 단계들을 제공해야 한다.
- 4. 보상을 받을 자격이 있을 경우 자금을 수령할 지갑 주소를 포함해야 한다.

범위 외(Out of Scope)

- 프롬 팀에 이미 알려졌거나 이전에 보고된 문제
- 프롬 위에 구축된 서드파티 애플리케이션의 취약점
- 실제 악용 경로가 없는 이론적 취약점
- 대규모 자원을 요구하는 DOS 공격

법적 및 윤리적 지침(Legal and Ethical Guidelines)

다음 규칙을 위반하면 보상의 일부 또는 전부를 상실할 수 있다:

- 문제를 입증하는 데 필요한 범위를 넘어 취약점을 악용하지 말 것

- 본인 소유가 아닌 데이터에 접근, 수정 또는 파괴하지 말 것
- 프롬 팀이 문제를 해결하기 전까지 취약점을 공개하지 말 것
- **심각도 오인:** 명백히 치명적이지 않은 버그를 치명적이라고 주장하지 말 것
- **범위 오인:** 범위 내 자산에 영향을 주지 않는 버그를 범위 내라고 주장하지 말 것
- 프로젝트의 명시적 서면 동의 없이 사용자 또는 프로토콜 자금을 보호하기 위한 화이트해킹 활동 금지
- 프로토콜 개발자에 대한 피싱 또는 기타 사회공학적 공격 시도 금지
- 프로젝트에 가스비 지급을 요구하지 말 것
- 개인적 특성에 기반한 공격 금지
- 폭력 위협 금지
- 동의 없이 개인 정보를 공개하거나 공개 위협을 하지 말 것
- 갈취/협박 또는 갈취/협박 위협 금지
- 이미 공개된 버그를 보고하지 말 것
- 특정 프로젝트에 대한 버그 리포트 또는 그 존재 자체를 수정 및 보상 전까지 공개하지 말 것
- 모호한 제목, 불충분한 정보, 재현 불가 단계로 이루어진 불완전한 버그 리포트 제출 금지
- 스팸/저품질 버그 리포트 제출, 버그 리포트가 아닌 정보 제출 금지
- 영어가 아닌 언어로 버그 리포트 제출 금지

프로그램 기간(Program Duration)

이 버그 바운티 프로그램은 지속적으로 운영되며, 범위와 보상 구조에 대해 주기적인 검토와 업데이트가 이루어진다.

프롬은 이 종합적인 버그 바운티 프로그램을 통해 전 세계 보안 연구 커뮤니티의 전문성을 활용하고자 한다. 이를 통해 생태계 내 다른 성공적인 플랫폼들이 확립한 모범 사례를 따르며, 자사 솔루션의 강건성과 보안을 한층 강화하는 것을 목표로 한다.

일반 FAQ (General FAQs)

이 문서는 프롬과 관련된 일반적인 질문에 답한다. 더 심층적인 정보는 전체 독스를 참고하라.

프롬 네트워크란 정확히 무엇인가?(What Exactly is Prom network?)

프롬은 폴리곤 CDK를 기반으로 한 고성능 확장형 네트워크이다. 프롬은 트랜잭션 증명을 선택된 정산 체인 (Settlement chain) 위의 추가 체인에 제출함으로써, 생태계 간 다리를 구축한다.

프롬의 주요 특징은?(Key Features of Prom?)

- **멀티체인 지원:** 프롬은 EVM 및 비-EVM 호환 체인을 포함한 여러 네트워크에 트랜잭션 증명을 제출할 수 있어 높은 수준의 탈중앙성과 100% 가동성을 보장한다.
- **효율성 및 확장성:** zkSNARKs를 통해 비용을 절감하는 동시에 확장성을 향상시킨다.
- **고속 트랜잭션:** 폴리곤 제로(Polygon Zero) 기술을 활용하여 탁월한 속도를 제공한다.
- **강력한 보안:** 이더리움의 보안을 계승하여 최고 수준의 데이터 무결성을 보장한다.
- **EVM 호환성:** 익숙한 EVM 환경에서 원활한 통합을 지원한다.

프롬 사용 시 가스비 절감 기대치는?(Gas Fee Reduction Expectations with Prom?)

사용자는 이더리움 레이어 1 대비 가스비를 크게 줄일 수 있다. 프롬의 프로토콜은 여러 트랜잭션을 배치 처리하여 단일 레이어 1 트랜잭션 비용을 여러 트랜잭션에 분산시킨다.

프롬이 트랜잭션에 별도 토큰을 사용하나? (Does Prom Utilize a Separate Token for Transactions?)

그렇다. 프롬에서의 트랜잭션은 **PROM 토큰**을 가스비로 사용한다. 또한 프롬은 ERC-4337 기반 계정 추상화를 지원하여, 사용자 관점에서 어떤 토큰으로든 투명하게 수수료를 지불할 수 있도록 한다.

프롬에서 배포에 적합한 디앱은? (Suitable dApps for Deployment on Prom?)

EVM 호환 dApp은 모두 프롬에 배포할 수 있다. 다만 현재 지원되지 않는 특정 프리컴파일(precompile) 컨트랙트를 요구하는 경우는 제외된다. 호환 가능한 프리컴파일 컨트랙트에 대한 상세 정보는 문서에서 확인할 수 있다.

프롬의 크로스체인 상호작용 기능은?(Prom Capability for Cross-Chain Interactions?)

현재 프롬은 크로스체인 기능을 지원하지 않는다. 그러나 향후 로드맵에는 여러 네트워크 간 자산 이동 지원이 포함되어 있으며, 런칭 시점에는 제공되지 않는다.

프롬의 주요 활용 사례는?(Principal Use Cases for Prom?)

- **디파이(DeFi) 애플리케이션:** 높은 보안성과 효율적 자본 활용이 가능하여 적합하다.
- **NFT, GameFi, 엔터프라이즈 애플리케이션:** 낮은 가스비, 빠른 트랜잭션 속도, 강력한 보안 덕분에 유리하다.
- **결제:** 최소 수수료로 실시간 거래를 원하는 사용자에게 이상적이다.

레이어 1에서 프롬 증명의 신뢰성은? (How Reliable are Prom Proofs on Layer 1?)

프롬 증명의 정확성은 **영지식 롤업 스마트 컨트랙트**에 의해 보장된다. 데이터 가용성과 SNARK 증명의 손쉬운 검증이 신뢰성을 뒷받침한다.

프롬은 Solidity 및 Vyper와 호환되는가? (Compatibility of Prom with Solidity and Vyper?)

그렇다. 프롬은 Solidity와 Vyper를 포함하여 EVM 연산 명령어로 컴파일되는 모든 언어를 지원한다.

프롬에서 RPC 노드란? (What Constitutes an RPC Node in Prom?)

프롬의 RPC(Remote Procedure Call) 노드는 이더리움과 호환되는 JSON-RPC 인터페이스로, 메타 마스크(Metamask), 이더스캔(Etherscan), Infura와 같은 기존 도구와의 원활한 통합을 가능하게 한다. 이는 geth와 같은 이더리움 노드의 작동 방식과 유사하다.

프롬의 JSON-RPC EVM 쿼리 사양 지원 여부는?(Prom Support for JSON-RPC EVM Query Specifications?)

프롬은 모든 표준 쿼리(eth* 엔드포인트)를 지원한다. 또한 debug*와 같은 추가 엔드포인트 지원을 확대하기 위한 작업이 진행 중이다.

프로토콜 FAQ(Protocol FAQs)

이 문서는 프롬 프로토콜과 관련된 일반적인 질문을 다룬다. 더 자세한 정보는 전체 독스를 참고하라.

프롬에서 트랜잭션은 어떻게 수집되고 정렬되는가? (How Are Transactions Collected and Ordered in Prom?)

- 사용자는 지갑에서 개인키를 사용해 트랜잭션을 생성하고 서명한다.
- 트랜잭션은 신뢰할 수 있는 시퀀서 노드의 JSON-RPC 인터페이스로 전송된다.
- 트랜잭션은 대기 중인 트랜잭션 풀에 저장되어 시퀀서의 검토를 기다린다.
- 신뢰할 수 있는 시퀀서는 풀에서 트랜잭션을 선택하고 실행 여부와 순서를 결정한다.
- 마지막으로 시퀀서는 이 트랜잭션들을 배치로 묶고 순서를 지정한다.

시퀀서 배치 생성에 특정 주기가 있는가? (Is There a Specific Interval for Sequencer Batch Creation?)

- 시퀀서는 열려 있는 배치를 유지하며, 트랜잭션이 가득 차거나 중요한 타임아웃이 발생할 때까지 트랜잭션을 추가한다.
- 배치가 128K에 도달하거나 주요 타임아웃이 발생하면, 시퀀서는 L1에 시퀀싱 트랜잭션을 전송한다.

L1에서 트랜잭션이 최종 확정되는 단계는? (Stages of Transaction Finalization on L1?)

- **Trusted State:** 신뢰할 수 있는 시퀀서가 즉시 제공하는 상태로, L1 트랜잭션이 필요하지 않다.
- **Virtual State:** L1에서 트랜잭션의 순서가 불변이며 최종 확정된다. 누구나 이 상태를 계산할 수 있다.
- **Verified State:** 가상 상태가 스마트 컨트랙트에 의해 확인되면, 자금 인출이 가능해진다.

시퀀서는 어떻게 트랜잭션을 검증하여 증명 생성을 준비하는가?(How Does a Sequencer Validate Transactions for Proof Generation?)

- 트랜잭션 형식과 필수 정보를 점검한다.
- 가스 수수료를 지불할 충분한 자금과 호출된 스마트 컨트랙트의 유효성을 확인한다.
- 트랜잭션이 중복되거나 이중 지불(double-spend)이 아님을 보장한다.

- 유효한 트랜잭션은 프롬의 현재 상태에 적용되어 스마트 컨트랙트 상태와 계정 잔고를 갱신한다.

트랜잭션은 언제 프롬에서 최종성을 얻는가?(When Do Transactions Achieve Finality in Prom?)

- **Trusted State 최종성:** 시퀀서가 트랜잭션을 시퀀싱하면 발생한다.
- **Virtual State 최종성:** 트랜잭션이 L1에 존재하고 최종 확정되면 달성된다.
- **Consolidated/Verified State 최종성:** 인출에 필요하며, 프루버가 암묵적 상태를 명시적 상태로 변환한 후 발생한다.

시퀀서와 프루버는 내부적인가, 외부적인가? 탈중앙화는 어떻게 보장되는가?(Are Sequencers and Provers Internal or External, and How is Decentralization Ensured?)

- 초기에는 프롬의 시퀀서와 프루버 모두 중앙화된다.
- 향후 계획에는 시퀀서를 탈중앙화하고 프루버를 위한 마켓플레이스를 구축하는 것이 포함된다.
- 프루버의 탈중앙화 시스템은 네트워크 전체의 탈중앙화를 위해 매우 중요하다.

zkNode가 시퀀서와 어그리게이터 역할을 동시에 수행할 수 있는가?(Can a zkNode Function as Both Sequencer and Aggregator?)

- 가능하다. 영지식 증명 프로토콜의 구현에 따라 zkNode가 두 역할을 모두 수행할 수 있다.
- 일부 프로토콜에서는 네트워크 보안과 효율성을 보장하기 위해 노드의 역할이 사전에 정해지기도 한다.

강제 배치(Forced Batches)란 무엇인가?(What Are Forced Batches?)

- 강제 배치는 L1 트랜잭션에 포함된 L2 배치로, 신뢰할 수 있는 시퀀서가 반드시 포함해야 하는 배치이다.
- 이 메커니즘은 사용자가 시퀀서에 의해 검열되더라도 자금을 인출할 수 있도록 보장하며, 검열 저항성을 제공한다.

