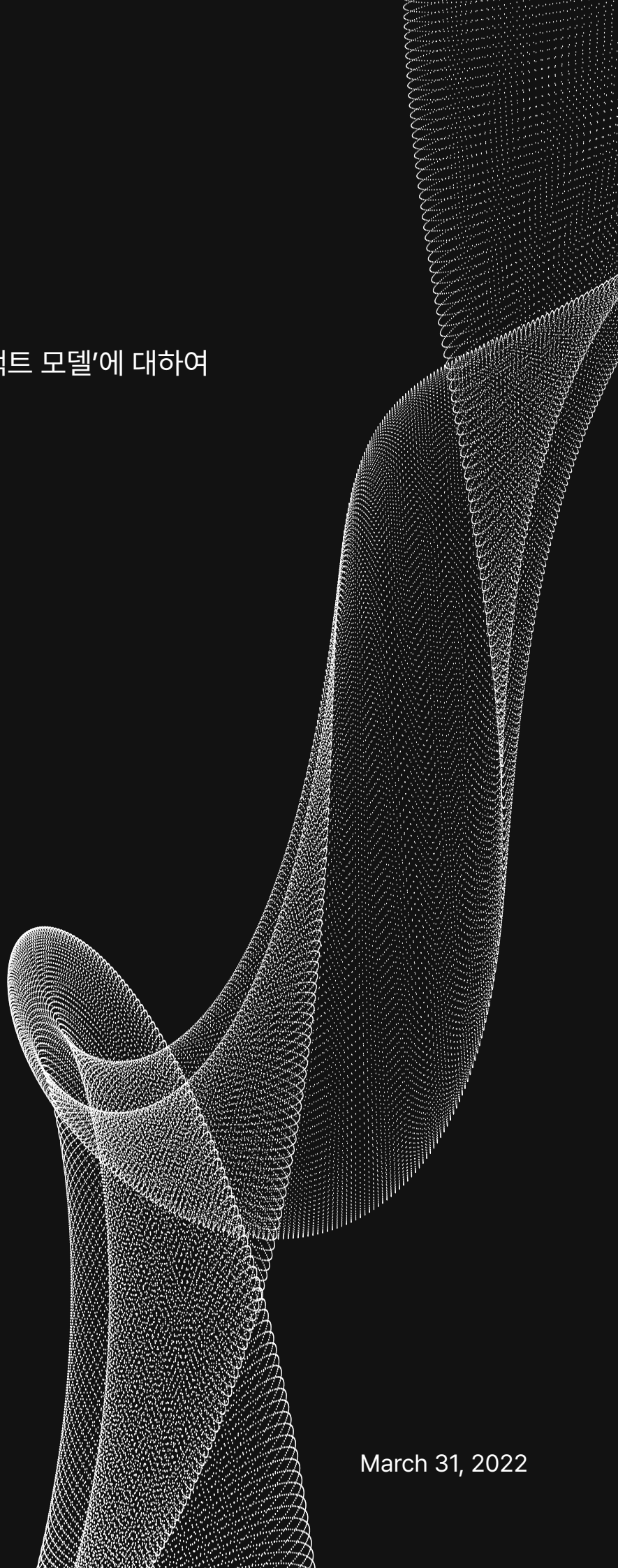
프로토콘 황서

1편 : 컨트랙트 모델

모델 기반 스마트 컨트랙트, '컨트랙트 모델'에 대하여



최윤일, 배민효

March 31, 2022

목차

1. 개요 3

2. 모델 3

2.1 모델 기반 개발 방법론 3

2.2 모델 기반 스마트 컨트랙트 4

3. 프로토콘 컨트랙트 모델 명세 5

3.1 단위 컨트랙트 모델 5

3.2 합성 컨트랙트 모델 7

4. 프로토콘 컨트랙트 모델의 특장점 8

4.1 모델 기반 개발 8

4.2 베어메탈 (Baremetal) 9

4.3 프로그래밍 언어 독립성 9

5. 결론 10

1. 개요

본 문서는 일반적인 블록체인의 스마트 컨트랙트(Smart Contracts)에 대응되는 프로토콘[1]의 '컨트랙트 모델(Contrat Model)'을 정의한다.
프로토콘의 컨트랙트 모델(이하 컨트랙트 모델)은 일반적인 블록체인의 스마트 컨트랙트와 유사한 개념으로, 현존하는 스마트 컨트랙트의 모든 기능을 수용하면서 동시에 이의 단점을 보완한 새로운 스마트 컨트랙트 개발 방법론이다. 이러한 새로운 접근 방법은 블록체인에 연동되는 수 많은 응용 시스템과 응용 서비스의 신뢰성(Reliability)과 보안성(Security)을 보장하는 기반이 된다. 또한, 컨트랙트 모델은 프로그래밍 언어 기반의 스마트 컨트랙트와 달리 성능과 블록체인 사용 비용 측면, 개발 효율성 측면, 보안성 측면에서도 탁월한 우수성을 제공한다.
본 황서는 컨트랙트 모델에 대한 소개와 더불어 모델 명세 및 특장점에 대해 설명한다. 이어지는 2장에서는 스마트 컨트랙트에 '모델' 개념 도입의 배경과 필요성을 제시하고, 3장에서는 세부 명세(specification)를 통해 '컨트랙트 모델'을 정의한다. 4장에서는 컨트랙트 모델의 특장점을 설명하고, 5장에서는 본 논고의 결론을 제시한다.

2. 모델

2.1 모델 기반 개발 방법론

모델 기반의 개발 방법론(Model-based Development 혹은 Model-based systems engineering)[2]은 높은 복잡도를 가지면서도 높은 신뢰성과 높은 보안성을 요구하는 시스템(예, 항공기, 군사무기 등)을 개발할 때 주로 적용하는 방법론이다.

최근 NFT 프로젝트나 디파이(DeFi) 프로젝트에 엄청난 규모의 자금이 모여들고 있다. 이러한 시장 상황은 매우 안정적이면서 가용성 높은 서비스 제공을 요구하고 있으며, 특히, 해킹 등의 보안 위협으로부터 안전을 보장 받을 수 있는 새로운 스마트 컨트랙트를 요구하고 있다. 즉, 이더리움이 제시한 VM(Virtual Machine) 구조보다 더 안정적이고 더 신뢰성 있고 더 효율적인 스마트 컨트랙트 개발 방법론이 필요한 시점이다. 프로토콘의 '컨트랙트 모델'은 이러한 요구에 부합할 수 있는 뛰어난 신뢰성과 높은 보안성을 제공하는 새로운 스마트 컨트랙트 개발 방법론이다.

모델 기반 개발 방법론에서는 모델을 수학적 명세를 통해 정의하고, 이 정의에 맞추어 기초가 되는 '기본모델(Unit Contract Model)'을 개발한다. 기본모델들은 신뢰성 및 보안성에 대한 엄격한 시험과 검증을 거친다. 이렇게 개발된 기본 모델을 조합하여 합성 모델을 만들 수 있으며, 이를 통해 신뢰성과 보안성이 검증된 다양한 응용 시스템과 서비스를 구축할 수 있다.

---

[1] 프로토콘(Protocon)은 프로토콜 기반으로 스스로 운영되는 디지털 경제를 구축하는 것을 목표로 하는 블록체인 프로젝트를 가리키며, PBFT 알고리즘과 스마트 컨트랙트를 새롭게 정의한 '미텀(Mitum)' 블록체인을 기반으로 한다.
[2] "Model-based systems engineering for aerospace," https://resources.sw.siemens.com/en-US/e-book-model-based-systems-engineering-aerospace.

2.2 모델 기반 스마트 컨트랙트

스마트 컨트랙트라는 개념은 1996년 닉 자보(Nick Szabo)에 의해 탄생[3]했지만, 스마트 컨트랙트라는 용어 및 개념이 널리 알려지게 된 계기는 이더리움(Ethereum)이라는 것은 부정할 수 없을 것이다.

이더리움은 2015년 최초 이더리움 네트워크가 가동을 시작한 이후 매우 안정적으로 블록체인 네트워크가 운영되고 있다. 이더리움을 뛰어 넘겠다고 천명한 솔라나, 클레이튼 등 대부분의 메인넷들이 과부하 내지는 서비스 거부 공격 등으로 다운이 되는 경우가 발생하고 있는 것이 반해 이더리움 메인넷은 매우 안정적으로 운영되고 있기 때문에, 이더리움의 암호화폐인 이더(Ether)는 현재까지 비트코인에 이어 시가총액 2위의 암호화폐로 자리매김 하고 있다.

물론, 이더리움 네트워크에서 대규모의 해킹 사건이 없었던 것은 아니다. 대표적으로 2016년 2월 1,500억원 규모의 다오(The DAO) 펀드가 해킹[4]되었다. 최근에는 유니스왑(Uniswap)과 디포스(dForce) 등 다수의 디파이(DeFi) 프로젝트에서 이더리움 네트워크와 관련된 해킹이 발생했다. 그런데, 이러한 해킹 사건들은 이더리움 메인넷 자체의 결함이 아니고 이더리움 스마트 컨트랙트의 결함에 의해 발생한 사건들이라는 사실에 주목할 필요가 있다.

이더리움의 스마트 컨트랙트는 솔리디티(Solidity)라는 언어를 사용하여 개발되는데, 솔리디티 언어는 대표적으로 아래와 같은 6대 취약점을 가지고 있다.[5]

1) 오버플로 및 언더플로 문제
2) 메시지 호출과 접근권한 제어 문제
3) 리엔트런시 문제
4) 짧은 주소 공격
5) 잔액 조건 무효화 공격
6) 도스 공격

이처럼 이더리움의 스마트 컨트랙트가 보안 취약점을 가지고 있다는 문제는 계속해서 제기되고 있다. 2018년 싱가포르 국립대학교(National University of Singapore)와 런던대학교(University College London)의 연구를 통해 이더리움 네트워크에서 3만 4,000개 이상의 취약한 스마트 컨트랙트가 발견되었다.[6] 이더리움 커뮤니티의 버그 바운티[7]에는 발견된 여러 취약점들이 지속적으로 보고되고 있다. 또한, IBM 리서치 연구원들이 제우스(ZEUS)라는 이름으로 발표한 논문[8]에 따르면, (그들의 분석기로 분석한 결과) 현재 배포된 스마트 컨트랙트 중에서 약 95%에 달하는 스마트 컨트랙트가 하나 이상의 취약점을 가지고 있다고 한다. 문제는 이러한 취약점이 취약점으로 끝나는 게 아니라, 작게는 수억에서 많게는 수천억 규모의 실제 해킹 사건으로 이어진다는 데에 있다.

---

[3] Szabo, Nick. "Formalizing and securing relationships on public networks," First Monday, 1997.
[4] 더 다오(The DAO) 해킹 사건 이후 이더리움은 이더리움 클래식으로부터 하드포크 되었다.
[5] "How to Secure Your Smart Contracts: 6 Solidity Vulnerabilities and how to avoid them," https://medium.com/loom-network/how-to-secure-your-smart-contracts-6-solidity-vulnerabilities-and-how-to-avoid-them-part-1-c33048d4d17d.
[6] "34,200 Ethereum Contracts Vulnerable to Hackers, Containing Millions in Ether," https://u.today/34200-ethereum-contracts-vulnerable-to-hackers-containing-millions-in-ether.
[7] "ETHEREUM Bounty Program," https://bounty.ethereum.org.
[8] Sukrit Kalra et al., "ZEUS: Analyzing Safety of Smart Contracts," IBM research, 2018.

스마트 컨트랙트의 보안성 문제는 비단 이더리움에 국한되지 않는다. 최근 다수의 NFT 프로젝트와 디파이 프로젝트를 운영중인 솔라나(Solana) 블록체인의 경우 러스트(Rust)라는 언어를 통해 스마트 컨트랙트를 개발하고 있다. 2022년 2월 2일 솔라나의 크로스체인 브릿지(Bridge) 서비스 웜홀이 해킹 당해 3,900억원 규모의 피해가 발생했는데, 해킹 분석 결과 해커는 패치가 되지 않은 러스트 스마트 컨트랙트를 악용한 것으로 나타났다.[9]

이러한 스마트 컨트랙트의 문제는 대단히 위험하고 심각한데, 그것은 데이터 위변조가 불가능하고 디지털 자산의 고유성과 소유권과 자산가치를 보장해 준다는 블록체인의 이상과 목표를, 다른 무엇도 아닌, 블록체인을 다양한 용도로 활용하기 위한 목적으로 만들어진 '스마트 컨트랙트'가 파괴하고 있기 때문이다.

이러한 상황을 개선하기 위해 우리는 보다 안정적이고 신뢰성 있는 스마트 컨트랙트를 제공하는 방법론으로, 항공기나 군사무기 등 높은 신뢰성과 높은 보안성을 요구하는 시스템 개발에 적용되고 있는 '모델 기반 개발 방법론'(Model Based System Engineering)을 블록체인 산업에 도입하고자 한다. 우리는 이것을 '컨트랙트 모델'이라 명명했다.

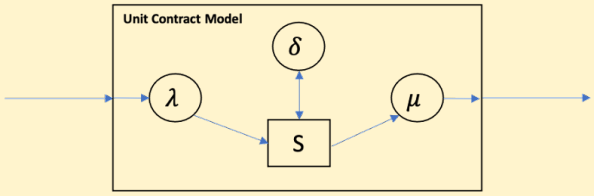
3. 프로토콘 컨트랙트 모델 명세

프로토콘의 컨트랙트 모델은 '단위 컨트랙트 모델 (Unit Contract Model)'과 '합성 컨트랙트 모델 (Composite Contract Model)' 두 가지로 정의된다. 단위 컨트랙트 모델은 더 이상 쪼갤 수 없는 기본 모델이며, 합성 컨트랙트 모델은 하나 이상의 단위 컨트랙트 모델을 조합하여 만들어 내는 모델을 말한다.

3.1 단위 컨트랙트 모델

$$M=\langle X, Y, S, \lambda, \delta, \mu \rangle$$

- X: a set of inputs
- Y: a set of outputs
- S: a set of states
- $\lambda: X \rightarrow S$, input transform function
- $\delta: S \rightarrow S$, state transition function
- $\mu: S \rightarrow Y$, output function



[그림 1] 단위 컨트랙트 모델 명세

단위 컨트랙트 모델 'M'은 3개의 집합과 3개의 함수로 구성된다.
3개의 집합은 입력과 출력, 그리고 모델의 상태를 나타낸다. 모델의 상태란 모델을 구성하는 데이터의 집합을 말한다

[9] "Solana's Wormhole Hack Post-Mortem Analysis," https://extropy-io.medium.com/solanas-wormhole-hack-post-mortem-analysis-3b68b9e88e13.

3개의 함수는 입력변환 함수, 상태변이 함수, 출력 함수로 구성된다. 입력변환 함수는 입력된 데이터(파라미터들)을 상태집합의 원소로 변환한다. 상태변이 함수는 모델의 상태 데이터에 대한 연산을 통해 상태를 바꾸는 연산을 수행하는 함수이다. 출력 함수는 상태 데이터에서 요구되는 데이터 내용과 형식에 따라 값을 출력해 주는 함수이다.

일반적인 모델의 동작은 입력변환 함수, 상태변이 함수, 출력 함수, 즉, 세 함수 쌍 (λ, δ, μ)의 연속동작으로 하나의 오퍼레이션(Operation)을 수행한다.

단위 컨트랙트 모델의 구조와 동작에 대한 이해를 돕기 위해 '미텀 커런시(Mitum Currency)'[10] 모델을 예를 들어 설명하도록 한다.

'미텀 커런시' 모델은 미텀 블록체인 상에서 사용할 토큰을 생성하고 어카운트(account)간 토큰을 전송하는 등의 오퍼레이션을 수행하는 단위 컨트랙트 모델이다.

미텀 커런시 모델의 상태집합은 생성된 토큰을 보유하고 있는 어카운트와 토큰 개수를 쌍으로 하는 데이터의 집합이다.

Account	Token Account	Difference
0xa374	258	0
0xb28d	27	0
0xdb8d	102	0

[그림 2] 미텀 커런시 상태집합 예시

가령, 어카운트 0xa374 에서 0xb28d 로 25개의 토큰을 전송하는 오퍼레이션을 수행하는 경우 입력 데이터는

$$x = (0xa374, 0xb28d, 25) \in X$$

이 되고, 입력변환 함수 λ의 수행 결과는 아래와 같다.

Account	Token Account	Difference
0xa374	258	-25
0xb28d	27	25
0xdb8d	102	0

[그림 3] 입력변환 함수 λ의 수행 후 상태집합

[10] "Mitum currency's documentation," https://mitum-currency-doc.readthedocs.io/en/latest/.

상태변환 함수 δ의 수행 결과는 아래와 같다.

Account	Token Account	Difference
0xa374	233	0
0xb28d	52	0
0xdb8d	102	0

[그림 4] 상태변환 함수 δ의 수행 후 상태집합

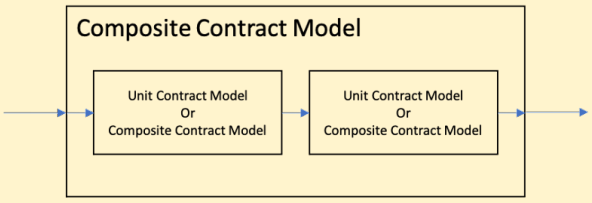
출력 함수 μ는 요구에 따라, 오퍼레이션 수행결과의 성공 여부 또는 변화된 어카운트의 토큰 카운트 등을 출력데이터로 정할 수 있다.

3.2 합성 컨트랙트 모델

$$C=\langle X, Y, \{M_i\}, \{C_j\}, R\rangle$$

- X: a set of inputs
- Y: a set of outputs
- $\{M_i\}$: a set of unit contract models
- $\{C_j\}$: a set of composite contract models
- R: $< Rxx, Rxy, Ryy >$, relations between contract models

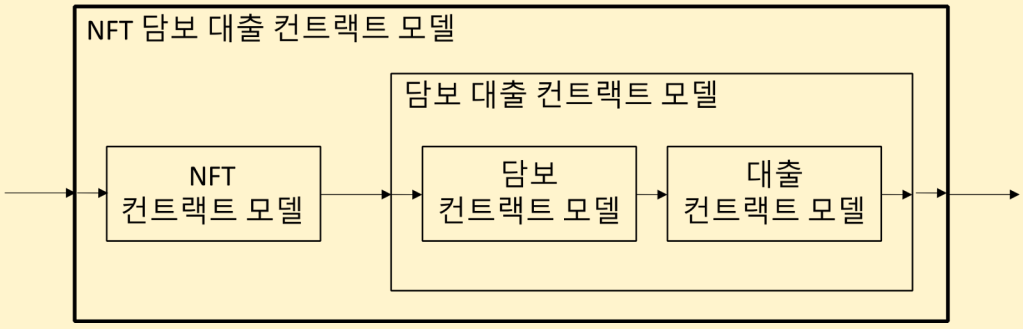
where
$$\begin{cases} Rxx \subseteq X \times \bigcup_{k=i\ or\ j} X_k \\ Ryx \subseteq \bigcup_{k=i\ or\ j} Y_k \times \bigcup_{l=i\ or\ j} X_l \\ Ryy: \bigcup_{k=i\ or\ j} X_k \rightarrow Y^{\Phi} \end{cases}$$



[그림 5] 합성 컨트랙트 모델 명세

합성 컨트랙트 모델 C는 4개의 집합과 3개의 관계쌍의 집합으로 구성된다. 4개의 집합은 입력과 출력의 집합, 합성모델을 구성하는 단위모델의 집합, 하위 합성모델의 집합이다. 3개의 관계쌍은 합성모델 외부에서 입력된 데이터를 내부 구성모델의 입력으로 매핑해주는 관계, 내부 모델간 입출력을 매핑해주는 관계, 내부 모델의 출력과 합성모델의 출력을 매핑해주는 관계로 구성된다.

NFT를 담보로 하는 담보대출 합성 컨트랙트 모델의 경우를 예를 들어 보자.



[그림 6] 합성 컨트랙트 모델 예시

'NFT 담보대출 컨트랙트 모델'은, NFT를 입력받아 모델 내 컨트랙트 어카운트에 NFT를 보유하는 'NFT 컨트랙트 모델'과, 담보를 바탕으로 대출을 발생시키는 '담보 대출 컨트랙트 모델'로 구성된다. 다시, '담보 대출 컨트랙트 모델'은 담보를 바탕으로 담보증권을 만들어 내는 '담보 컨트랙트 모델'과 담보증권을 바탕으로 대출 결과에 따라 암호화폐를 대출자에게 전달하는 '대출 컨트랙트 모델'로 구성된다. 그리고, 'NFT 담보 대출 컨트랙트 모델'은 위 그림에서 화살표로 표시되는 입출력 관계를 포함한다.

이처럼 컨트랙트 모델에서는 충분히 검증된 단위 컨트랙트 모델들을 조합하여, 시장 및 고객의 요구사항을 충족시키는 다양한 합성 컨트랙트 모델을 만들어낼 수 있다.

4. 프로토콘 컨트랙트 모델의 특장점

이제 프로토콘 컨트랙트 모델의 특징 및 장점에 대해 알아보자.

4.1 모델 기반 개발

대다수 블록체인의 스마트 컨트랙트는 별도의 프로그래밍 언어를 통해 작성된다. 이더리움의 경우 솔리디티, 솔라나는 러스트, 카르다노는 하스켈 프로그래밍 언어를 사용한다. 이러한 구조로 인해 소프트웨어 개발 전문가라 하더라도 스마트 컨트랙트를 개발하기 위해서 블록체인 별로 지정된 언어를 새롭게 습득해야 하며, 개발자가 해당 언어를 학습하고 일정 수준의 개발이 가능하기까지는 어쩔 수 없이 여러 시행착오를 겪게 마련이다. 이러한 과정에서 다양한 휴먼 에러가 발생하게 되고, 따라서 신뢰성 문제와 보안 취약성이 발생하게 된다. 위에서 언급한, 이더리움 스마트 컨트랙트의 취약점들 중 상당 부분은 이와 같은 과정 속에서 만들어진 것이다.

이에 반해 프로토콘의 모델 기반 스마트 컨트랙트, 즉, 컨트랙트 모델은 이미 검증된 단위 컨트랙트 모델을 구성요소로 하여, 마치 레고 블록을 이용하여 다양한 완성품을 만드는 것과 같이, 단위 모델들을 하나 하나 쌓아나가는 방식으로 개발할 수 있다. 새로운 단위 컨트랙트 모델이 필요한 경우, 모델의 명세, 즉, <X, Y, S, λ, δ, μ> 의 여섯 개의 구성요소를 정의하여 개발함으로써 프로그래밍 언어에 종속되지 않게 개발할 수 있다.[11] 뿐만 아니라, 미숙하거나 잘못된 프로그래밍 언어의 사용으로 인한 보안 취약성 발생 소지도 상당 부분 예방할 수 있다.

[11] 향후 개발될 GUI환경을 이용하면 별도의 프로그래밍 언어 없이도 컨트랙트 모델을 개발할 수 있다.

이렇게 개발된 컨트랙트 모델이 블록체인에 배포되기 위해서는, 블록체인의 합의 그룹을 구성하는 노드들의 검증 및 합의 과정을 거치게 된다. 이는 무분별한 컨트랙트 모델의 배포를 방지하고 개발된 컨트랙트 모델은 검증을 통한 신뢰성을 확보하도록 할 수 있다. 결국, 충분히 검증된 모델을 사용하여 스마트 컨트랙트가 배포 운영되므로 신뢰성과 보안성이 높아지게 된다.

이더리움이 제시한 스마트 컨트랙트는 그 구조상 동일한 기능을 하는 컨트랙트들이 프로젝트별로 중복으로 다수의 블록에 저장되고 실행되지만, 컨트랙트 모델에서는 동일한 기능을 필요로 하는 컨트랙트들은 동일한 단위 모델을 호출하게 되므로 전체 컴퓨팅 자원을 효율적으로 사용할 수 있다. 무엇보다 코드의 무분별한 하드포크에 따른 보안 취약성을 방지할 수 있다.
더 나아가 이러한 단위 모델들이 충분히 개발되고 나면 이것을 GUI 형태로 제공하여, GUI 툴을 활용해 다양한 기능들을 조합하여 보다 복잡한 서비스를 만들도록 하는 것도 가능하다. 이 단계까지 이르게 되면 웬만한 블록체인 응용 기능들은 복잡한 프로그래밍을 거치지 않고 GUI 툴만으로 개발될 수 있을 것이다.

이와 같은 새로운 개발 방법론을 통해 우리는 블록체인을 보다 쉽게 그리고 안전하게 사용하도록 하자는 블록체인 산업계의 숙원을 풀어내고자 한다.

4.2 베어메탈 (Baremetal)

프로토콘 컨트랙트 모델의 또 다른 차별점은 가상머신을 사용하지 않고 하드웨어(Baremetal)에서 직접 실행한다는 점이다. 일반적으로 블록체인 상에서 스마트 컨트랙트를 실행하기 위해서는 블록체인 별로 정의한 각각의 가상머신을 사용해야 한다. 이더리움의 경우는 EVM (Ethereum Virtual Machine)[12], 솔라나는 LLVM (Low Level Virtual Machine), 카르다노는 KEVM (K Ethereum Virtual Machine) 등 각각 자신들만의 가상머신을 사용한다.

이더리움, 솔라나, 카르다노 등에서는 각각의 스마트 컨트랙트 프로그램 언어로 코딩을 하고 컴파일을 하면 바이트 코드 형태의 머신코드(가상머신 인스트럭션)로 변환이 되고, 이 머신코드들은 에뮬레이터에 의해 각각의 가상머신에서 인스트럭션 단위로 실행된다. 잘 알려져 있다시피 에뮬레이터는 하드웨어 CPU의 동작을 소프트웨어로 흉내내어 동작시키는 것으로, 하드웨어 자체로 동작하는 경우에 비해 매우 느리다.

반면 컨트랙트 모델은 가상머신 위에서 수행되는 것이 아니라 하드웨어에서 직접 수행되므로 VM을 사용하는 타 블록체인의 스마트 컨트랙트들에 비해 탁월한 성능을 발휘한다. 즉 컨트랙트 모델은 가상머신을 사용하지 않아 하드웨어의 성능을 그대로 활용할 수 있다. 이것만으로도 우리는 블록체인의 처리 성능을 획기적으로 개선할 수 있게 된다.

4.3 프로그래밍 언어 독립성 (Independence of Programming Language)

프로토콘의 모델 컨트랙트는 프로그래밍 언어의 종속성 없는 스마트 컨트랙트 개발을 가능하게 한다.

---

[12] "ETHEREUM VIRTUAL MACHINE (EVM)," https://ethereum.org/en/developers/docs/evm/.

만약 누군가 새로운 컨트랙트 모델을 개발한다면, 그는 먼저 모델 명세, 즉, <X, Y, S, λ, δ, μ>를 정의하고, 정의된 명세에 따라 자신이 선호하는 프로그래밍 언어를 사용하여 구현하면 된다. 특정 언어에 충분히 숙달된 개발자라면 해당 언어의 특징과 보안 유의점 등을 잘 알 것이기에, 새로운 언어를 공부해서 개발하는 것보다 훨씬 효율적이고 안전하게 프로그램을 구현할 수 있다. 이는 프로토콘 컨트랙트 모델이 반드시 특정 언어를 사용해야만 하는 스마트 컨트랙트와 차별되는 매우 중요한 특징이다.

5. 결론

앞서 언급한대로, 최근 NFT 프로젝트나 디파이(DeFi) 프로젝트에 엄청난 규모의 자금이 모여들고 있다. 블록체인은 자산 성격의 디지털 데이터를 다룬다는 측면에서 신뢰성과 안전성을 확보하는 것이 다른 무엇보다 중요하다. 이러한 상황에서 매우 안정적이면서 가용성 높은 서비스 제공이 요구되고, 특히, 해킹 등의 보안 위협으로부터 안전을 보장 받을 수 있는 스마트 컨트랙트가 요구되고 있다.

이에 대해 프로토콘은 컨트랙트 모델이라는 개념을 제시한다. 이미 우리는 토큰 모델, DID 모델, 데이터 모델, NFT 모델 등 현재 블록체인 산업에서 사용하고 있는 대부분의 기능들을 모델 기반 스마트 컨트랙트 개발 방법론으로 성공적으로 개발해서 테스트넷에서 테스트를 완료 했으며, 프로토콘 네트워크 기반의 다양한 프로젝트에 적용 중에 있다. 우리는 블록체인을 필요로 하는 업계의 다양한 요구사항들을 적극적으로 수렴하면서 보다 풍부한 기능들을 지속적으로 추가할 예정이다.

결론적으로, 우리는 치명적인 단점을 가지고 있는 VM 기반의 스마트 컨트랙트를 극복하는 대안으로 프로토콘 컨트랙트 모델을 제시한다. 이는 높은 신뢰성, 강화된 보안성, 빠른 처리성능이 요구되는 프로젝트에 최적의 기술적 기반을 제공할 것이다.

프로젝트 정보

홈페이지 : https://protocon.io/

깃허브 : https://github.com/ProtoconNet

이메일 : contact@protocon.io