

Hyperledger Fabric 기반의 지능형 IoT MES(Manufacturing Execution System) 플랫폼 구현

남 기 훈*

Implementation of Intelligent IoT MES Platform based on Hyperledger Fabric

Ki-Hun Nam*

This work was supported by Seokyeong University in 2020.

요 약

최근 몇 년 동안 지능형 IoT 시스템은 생산되는 데이터의 분석 및 모니터링을 통해 생산 효율성을 극대화한다. 또한 시장 상황에 따라 재고 및 생산량을 조정하여 재고 손실을 줄일 수 있는 기능이 있다. 대부분의 MES 시스템이 가동 중이기 때문에 제품 개략도 및 성능과 같은 정보도 포함된다. 보안이 철저하기 때문에 일반 보안 시스템보다 성능이 우수한 보안 시스템이 필요하다. 본 논문에서는 기존 블록체인에서 PKC(공개키 인증서)를 속성화하기 위해 정의한 AC(속성 인증서)를 보완하는 Hyperledger Fabric 기반 RBAC(Role-Based Access Control)를 제안한다. 엔티티의 역할에 대한 액세스 및 권한을 빠르고 정확하게 식별하고 반영할 수 있는 Hyperledger Fabric 기반 RBAC-PAC 모델을 제안한다. 제안된 시스템의 성능은 효율성 측면에서 기존 연구와 비교된다. 이 작업을 통해 제안된 시스템이 더 많은 엔티티를 갖고 더 자주 권리를 갱신할수록 더 효율적이라는 것을 발견하였다.

Abstract

In recent years, intelligent IoT systems have maximized production efficiency through analysis and monitoring of the data they produce. In addition, it has features that can reduce inventory losses by adjusting inventory and output according to market conditions. Because most of the MES system is in operation, it also contains information such as product schematics and performance. As security is strict, a security system is needed that performs better than a typical security system. This paper proposes a Hyperledger Fabric based Role-Based Access Control(RBAC) that complements the Attribute Certificate(AC), which was defined for attributing Public Key Certificates(PKC) in the existing blockchain. We propose a Hyperledger Fabric based RBAC-PAC model that can quickly and accurately identify and reflect access and authorization to the roles of entities. The performance of the proposed system is compared with existing research in terms of efficiency. Throughout this task, we find that the proposed system has a higher number of entities and the more frequent renewal of rights, the more efficient it is.

Keywords

MES(Manufacturing Execution System), blockchain, hyperledger fabric, security, IoT

* 서경대학교 컴퓨터공학과 조교수
- ORCID: <https://orcid.org/0000-0002-0724-912X>

• Received: Aug. 20, 2021, Revised: Sep. 14, 2021, Accepted: Sep. 17, 2021
• Corresponding Author: Ki-Hun Nam
Dept of Computer Engineering, SeoKyeong University, 4-ding, Jeongneung, Seongbuk-gu, Seoul, Korea
Tel.: +82-2-940-7667, Email: namkh@skuniv.ac.kr

1. 서 론

최근 4차 산업혁명의 시대 진입으로 인하여 제조업 분야의 스마트팩토리 도입에 대한 관심이 높아지고 있다. 스마트팩토리는 생산 과정에 필요한 전체 사물들을 산업 사물인터넷 기술을 통해 연결하여 통신체계를 구축하고 빅데이터, 인공지능, 클라우드 등 다양한 ICT(Information & Communication Technology) 기술들이 적용되면서 제조 생산의 자동화 및 최적화를 구현하고 있다. 이러한 스마트팩토리 시스템은 기존 노동력 중심의 생산 공정들이 지능화 및 자동화되어 운영됨으로써 제조에서 발생하는 불량률 감소와 생산성 개선 등의 다양한 결과를 가져오고 있으며 다양한 비즈니스 연계를 통해 새로운 부가가치를 창출하고 있다. 4차 산업혁명의 목적은 자원의 효율성을 높이고, 가치사슬에서 고객과 공급자를 위한 지능적이고 스마트한 공장 구현으로 정의하고 있으며, 핵심기술로는 사이버물리시스템(Cyber physical system)과 IoT(Internet of Things) 그리고 사이버 보안 기술이 있다[1][2].

4차 산업혁명 시대의 스마트팩토리는 기존의 공장자동화(FA, Factory Automation) 수준을 넘어선 차세대 디지털 신기술과 제조기술이 접목된 소비자 중심의 지능화된 공장을 의미한다. 한 생산라인에서도 다양한 제품 생산이 가능하며 모듈화를 통해 대량맞춤에서 개인별 유연생산 체계로 변화할 것으로 보고 있다. 스마트팩토리로의 전환은 제조업의 생산성을 획기적으로 향상시킬 것으로 전망되며 에너지 절감, 인간 중심의 작업환경 구현 또한 가능하다. 가상의 공간에서 제조현장을 모니터링 할 수 있을 뿐만 아니라 제어까지 가능하여 공장 관리가 용이하며 품질 및 원가 경쟁력 강화로도 이어질 것으로 전망된다[3].

모든 사물들이 연결된 스마트팩토리의 경우, 설계단계에서부터 최종 고객에게 전달되고 고객에게 서비스 되어지는 과정까지 수많은 정보들이 연결되어 공유되고 상호 유기적으로 결합되어 제공됨에 따라 제품 및 공정의 일부 정보 노출일지라도, 플랫폼에서 연결된 속성으로 인해 플랫폼 위에 축적된 전반적인 연계정보까지 유출될 위험이 커진다. 따라

서 4차 산업혁명 시대에는 사물인터넷 및 플랫폼의 보안기술이 센서나 네트워크 기술만큼이나 강조되고 있으며 기업들은 산업현장의 보안을 위협하는 요소들을 파악하고 체계적이고 치밀한 사이버 보안 전략을 수립해야 한다.

이에 본 논문에서는 4차 산업혁명 시대의 스마트팩토리 보안 이슈를 해결하기 위하여 블록체인 기반의 MES(Manufacturing Execution System) 플랫폼을 제안하였다. 본 논문의 구성은 Hyperledger Fabric 기반의 지능형 IoT MES 플랫폼을 제안하여 스마트팩토리 시스템의 보안성을 강화하고자 한다. 2장에서는 스마트팩토리 보안 이슈를 해결하기 위한 기존 관련 연구들을 정리하였다. 3장에서는 제안하는 Hyperledger Fabric 기반의 지능형 IoT MES 플랫폼을 설계하고, 4장에서는 구현된 시스템을 검증하고 효율성을 비교하여 결론을 맺고자 한다.

II. 관련 연구

2.1 스마트팩토리 : MES

스마트팩토리란 단순 자동화 공장이 아닌, 제품의 기획·설계·생산·유통·판매 등 전 과정을 ICT(정보통신기술)로 통합해 스스로 데이터를 수집하고 작업 명령을 내릴 수 있도록 설계된 ‘지능화 공장’을 의미한다. 지능화 공장은 사물인터넷(IoT)을 통해 실시간으로 기계의 상태나 공정의 진행상황 등의 정보를 수집하고 정보를 공유하며, 필요한 의사결정을 내리는 동시에 최고의 생산효율을 낼 수 있도록 한다[4][5].

스마트팩토리는 기존 제조기술에 인공지능, 사물인터넷, 클라우드(Cloud), 빅데이터(BigData), 정밀 등 다양한 ICT기술과의 융합을 통해 구축되며, 또한 높은 수준의 자동화 및 지능화된 인프라를 제공함으로써 생산성 향상과 더불어 에너지 절감 그리고 안전한 제조환경 구현이 가능하여 유연한 생산체계의 구축이 가능하다. 이를 통해 개인 맞춤 제조, 제조 서비스 융합 등 새로운 비즈니스 환경에 능동적 대응이 가능하다[6].

스마트팩토리는 기존의 자동화 공장과는 몇 가지

점에서 다른 차이점을 가지고 있다. 자동화 공장은 사전에 프로그래밍 된 명령에 따라 제품을 생산하고, 공정에 따른 변경이 가능하지만, 스마트팩토리는 실시간적으로 자율 생산이 가능하므로 설비, 재료, 환경 등의 상태에 따라 네트워크를 통해 스스로 판단하고 수행한다[7][8].

2.2 블록체인

블록체인은 관리 대상 데이터를 '블록'이라고 하는 소규모 데이터들이 P2P 방식을 기반으로 생성된 체인 형태의 연결고리 기반 분산 데이터 저장 환경에 저장하여 누구라도 임의로 수정할 수 없고 누구나 변경의 결과를 열람할 수 있는 분산 컴퓨팅 기술 기반의 원장 관리 기술이다. 이는 근본적으로 분산 데이터 저장기술의 한 형태가 지속적으로 변경되는 데이터를 모든 참여 노드에 기록한 변경 리스트로서 분산 노드의 운영자에 의한 임의 조작이 불가능하도록 고안되었다.

이러한 기능을 가능하게 하는 블록체인의 특징은 크게 두 가지로 살펴볼 수 있다[9]. 첫째는 시간별로 블록이 정리 되어진다. 한 블록에는 앞의 블록과 뒤의 블록을 연결하는 연결정보가 포함되어 있으며, 앞 블록의 내용을 변경하면 뒤에 이어지는 모든 블록을 다시 생성해야 한다. 따라서 과거 블록의 내용을 조작하는 것은 어렵다. 반대로 과거의 거래 기록이 존재한다면 그것은 그 시에 거래가 이루어졌다는 것을 객관적으로 알 수 있고 이러한 특징은 변조의 공격을 막아 무결성을 제공한다. 두 번째 중요한 특징은 분산원장이다. 블록체인 네트워크에 참가한 모든 사람이 모든 거래 기록을 기록한 원장을 소유하기 때문에 거래의 투명성이 높다는 것이 특징 하나이며 거래를 처리하는 중앙 시스템이 없어 탈중앙화가 가능하다.

블록체인에 참여하는 사용자들은 거래를 작성하고 자신의 개인키로 거래에 서명한다. 작성된 거래들은 다른 사용자들에게 브로드 캐스팅하여 전달하고, 합의과정에서 특정 합의 알고리즘을 통해 거래들을 하나의 블록으로 생성한다. 생성된 블록은 기본 블록체인에 연결되게 되고, 블록의 정보는 다른

사용자들에게 브로드 캐스팅한다. 사용자들은 블록체인에 연결된 블록의 정보들을 바탕으로 중앙 시스템 없이 사용자들 간에 데이터의 신뢰성을 확인할 수 있다[10]-[12].

2.3 Hyperledger Fabric

그림 1은 리눅스 재단에서 주관하는 블록체인 오픈소스 프로젝트이다. 금융, IoT, 물류, 제조, 기술 산업 등 여러 산업에 걸쳐 응용 가능한 블록체인 기술을 만드는 것이 목표로 하고 있다. Hyperledger 이외에도 R3, Ripple, Ethereum 등 다른 블록체인 플랫폼도 있다. 이 중 Hyperledger가 특별한 이유는, 프라이빗 블록체인 플랫폼으로서 기업 비즈니스를 구현하기에 적합한 환경이라는 점, 특정 비즈니스 모델에 특화된 타 플랫폼과 달리 여러 산업에 범용적으로 도입 가능한 기술 표준을 제시한다는 점이다.

Hyperledger는 위와 같은 차별화 전략을 구현하기 위해 다양한 기업용 블록체인 기술을 양산한다. 여기에는 분산 원장 프레임워크, 스마트 컨트랙트 엔진, 클라이언트 라이브러리, 그래픽 인터페이스, 기타 유틸리티, 샘플 어플리케이션 등이 포함된다. 이런 Hyperledger의 엮브렐라 전략(Umbrella strategy)은 공통 기반 요소를 재활용하도록 하여 커뮤니티를 강화하는 동시에 분산 원장 기술 요소의 빠른 발전을 유도하고 있다.

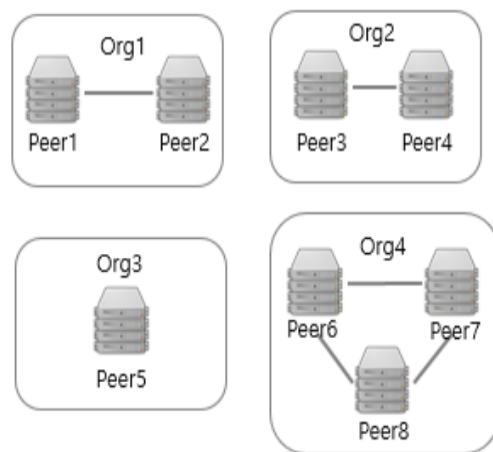


그림 1. Hyperledger 패브릭 조직 개념도
Fig. 1. Organizational chart of Hyperledger fabric

인큐베이팅 프로젝트는 크게 두 가지로 나눈다. 하나는 Hyperledger Framework이고, 다른 하나는 Hyperledger Tool이다.

2.3.1 Hyperledger Fabric 분산원장

그림 2와 같이 Hyperledger Framework의 핵심인 분산원장이다. 첫 번째로 Hyperledger Fabric의 원장은 현재 상태를 나타내는 World state, 원장의 생성 시점부터 현재까지의 사용기록을 저장하는 블록체인 두 가지로 구분할 수 있다. World state는 분산원장의 현재 값을 나타내고 블록체인은 생성 시점부터 현재까지의 모든 거래기록을 나타낸다.

두 번째로 World state에 저장된 데이터는 합의 과정에 의해 블록체인에 포함되기 전까지 체인 코드를 통해 조회/변경/삭제가 가능하며 단 여기서 합의에 의해 결정된 블록 및 블록체인은 절대 수정할 수 없다(Non-deterministic). World state는 데이터의 기록, 수정, 읽기 등이 빈번하게 발생하기 때문에 분산 데이터베이스로 구축해야 한다.

블록체인은 데이터 요청이 없고, append-only 방식의 저장이 목적이기 때문에 파일시스템 형태로 저장하면 된다.

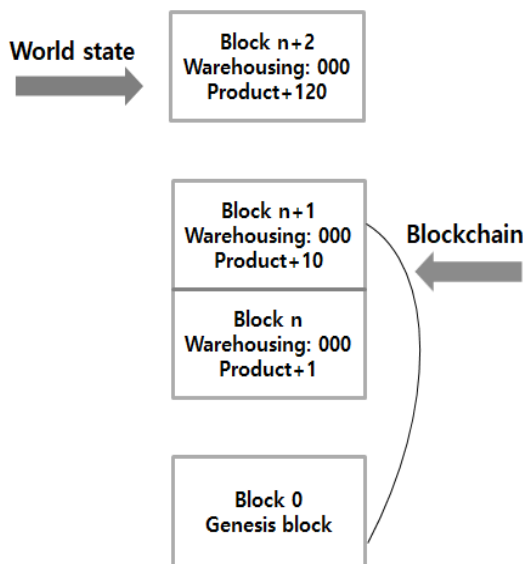


그림 2. 분산원장 구조
Fig. 2. Distributed ledger structure

2.3.2 Hyperledger Fabric MSP

두 번째로 Hyperledger Framework 중 MSP는 Hyperledger 중에서 Identity 기술을 바탕으로 한 Hyperledger Fabric 멤버십 관리 기술이다. 이 MSP를 이용하여 Peer, Orderer, Fabric-CA, Admin 등의 역할과 소속, 권한 등을 정의하는 중요한 기술이다. MSP를 통해서 Hyperledger Fabric의 구조 설계가 가능하며 각 조직의 특성과 용도에 맞게 조직을 세분화하여 MSP를 생성한 후 인증 시스템을 구축할 수도 있다. MSP에서는 크게 두 가지 볼 수 있는데 Local MSP와 Channel MSP이다.

본 논문은 연구에서 MES 플랫폼 구현할 때 가장 Hyperledger 중에서 MSP로 구성의 역할이 큰 Local MSP와 Channel MSP를 그림 3과 같이 구성하였다. Local MSP를 통해 어떤 노드가 Peer, Orderer 혹은 Client인지 정의할 수 있는 기능을 가지고 있으며 Channel MSP는 채널 구성원들에 대한 멤버십 정의와 권한을 부여할 때 사용된다. 채널에 참여한 구성원들은 각자의 Local MSP를 통해 하나의 Channel MSP를 생성하게 되는데 어떠한 조직에서 채널에 참여하려 할 때 채널 구성원은 Channel MSP를 참고하여 보증 또는 거절을 할 수 있다.

III. 제안 시스템

3.1 Hyperledger Fabric - MES 네트워크 관리 기능 설계

본 논문에서는 Hyperledger Fabric에서 보통 2개 이상의 MES간의 협정을 맺을 수 있게 네트워크를 구축하려 한다. 먼저 각 MES간 인증 설정을 해야 하는데 채널을 생성하여 비즈니스 네트워크를 구축할 수 있도록 설계를 작성한 후, MES와의 암호 인증 서비스를 구축해야 하는데 그림 4와 같이 3개의 컨소시엄을 맺은 Hyperledger Fabric 네트워크를 보여주고 있다.

본 논문은 해당 MES에는 2개의 채널을 통해 각각 다른 MES 네트워크가 구축되어 있다. Org1은 채널1에 참여, Org2는 채널 2에 참여, 그리고 Org3은 채널 1과 채널 2에 모두 참여가 되어있다.

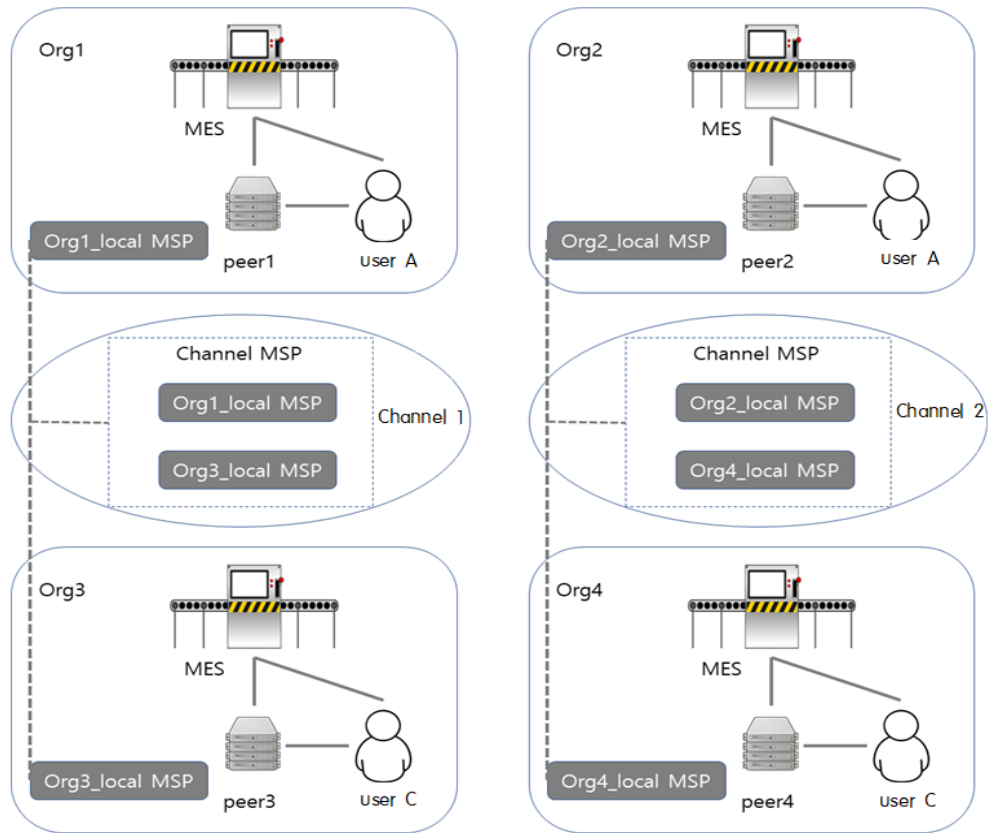


그림 3. Local MSP, Channel MSP 생성구조
 Fig. 3. Local MSP, channel MSP generating structure

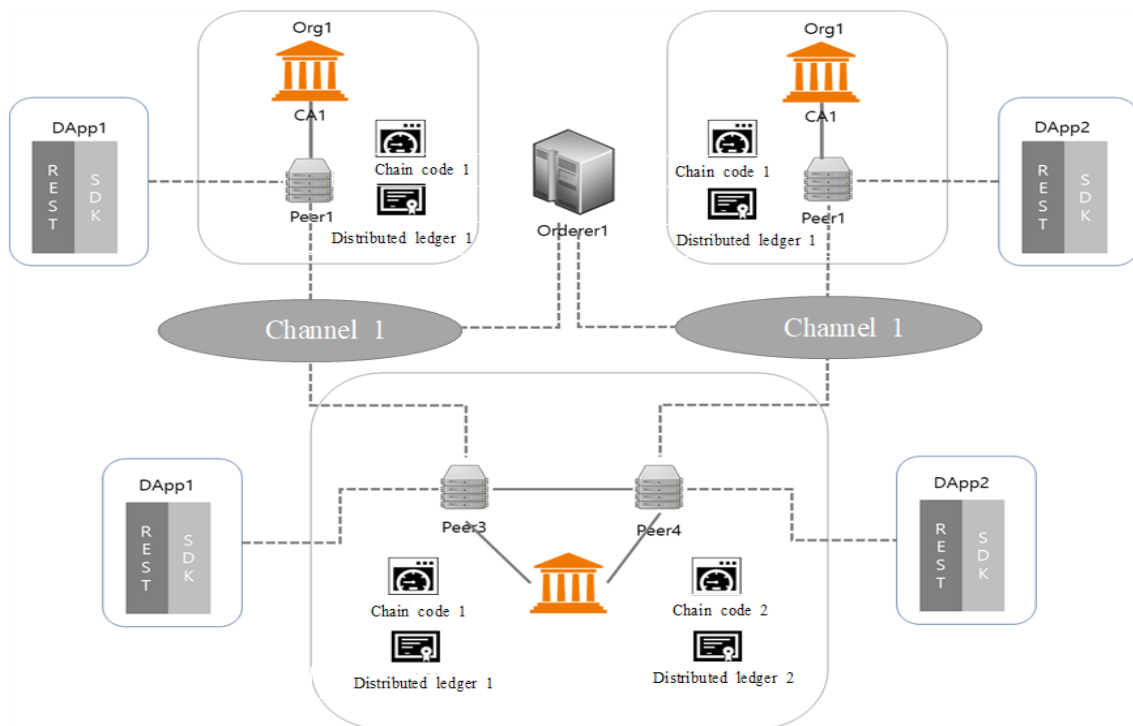


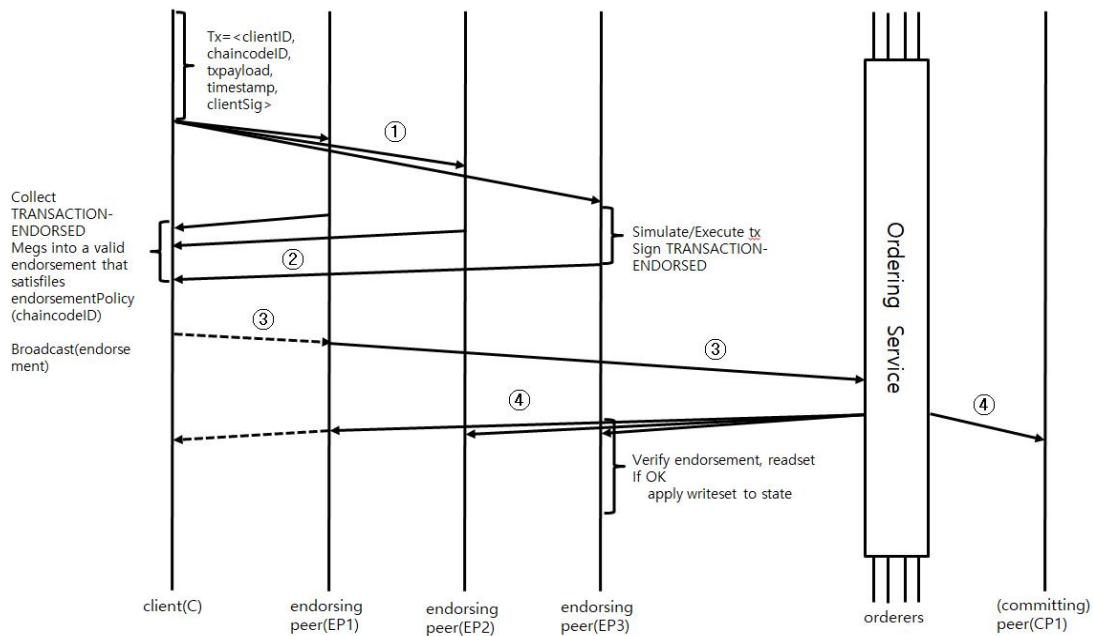
그림 4. MES 네트워크 구축 완성도
 Fig. 4. MES network construction completeness

MES에 암호 인증 서비스 하에 오더링 서비스 노드를 구축한 후 향후 추가될 peer, 채널, 클라이언트, 네트워크 정책, 채널 정책 등을 오더링 서비스에 저장된 Configuration block을 통해 설정할 수 있게끔 실행한다.

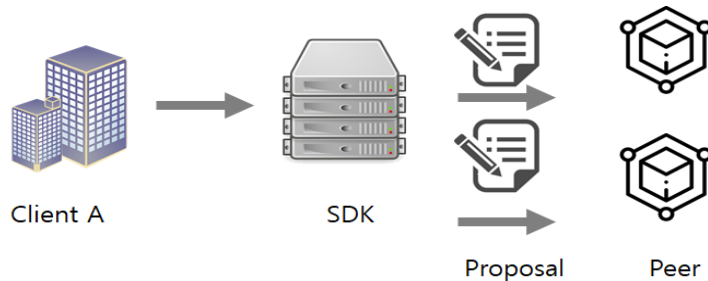
Org1과 Org3은 자신의 데이터 센터에 Peer를 설치한 후 채널 1에 참여시켜 Channel 1에 참여한 Peer는 채널 1에서 사용되는 분산원장 1을 자신의 로컬 저장소에 저장하고 이를 통해 비즈니스 데이터를 공유할 때만 실행한다. 암호 인증을 구축하려면 Endorsing Peer가 필요한지 확인을 통해 Endorsing Peer를 생성한다.

3.2 Hyperledger Fabric - MES Transaction

본 논문의 Transaction 흐름의 처리과정은 중요하다. 암호 인증 서비스가 작동이 되려면 그림 5와 같이 Client A가 블록체인 네트워크에 제고물품을 입고하는 요청을 보낸다고 가정하고 요청은 Client A와 Client B를 대표하는 A와 Peer B를 대상으로 하는데 보증 정책(Endorsement policy)에 두 Peer가 모든 Transaction을 보증해야 하므로 이 요청은 Peer A와 Peer B에 이동하게 하고, Transaction 제안이 작성이 필요하게 되는데 그 제안에는 노드, 자바, 파이썬을 활용하는 애플리케이션은 Transaction 제안을 생성하는 사용 가능한 API들 중 하나를 활용할 수 있게 설계를 했다.



(a) MES 트랜잭션 흐름



(b) Client A의 보증정책 전달 과정

그림 5. Hyperledger Fabric 네트워크 관리 기능

Fig. 5. Hyperledger Fabric Network management function

(a) MES transaction flow processing, (b) Client A's warranty policy delivery process

3.2.1 Endorsing Peer 서명 작업 Transaction 실행

그림 6에서 Endorsing Peer가 Transaction을 전달을 받게 되면 Transaction의 형식에 맞게 내용이 잘 채워졌는지 확인을 실행 작업 후, 이전에 제출된 적 있었던 Transaction 이력 조회, Transaction을 제출한 Client가 권한이 있는지 확인 작업을 하면, 확인이 된 Transaction Proposal을 인자로 받아서 체인코드를 실행하게 된다. State DB의 값에 체인코드가 실제로 실행 되었고 그 결과 값이 Read/Write Set을 반환한다. 그리하여 값 세트들은 보증 Peer의 서명과 함께 어플리케이션의 페이로드를 분석하는 ‘제안 응답’으로 반환하도록 어플리케이션에 그 결과 값이 저장 되도록 구축한다.

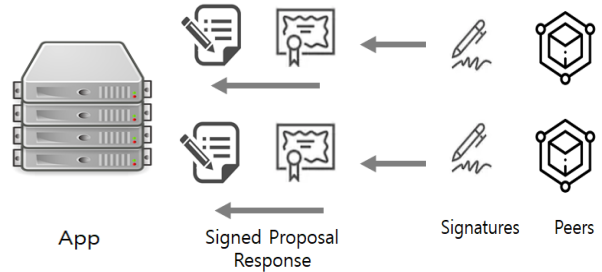


그림 6. 피어 서명 트랜잭션 전달
Fig. 6. Peer signed transaction propagation

3.2.2 Proposal Response 점검

그림 7처럼 어플리케이션은 보증 피어 서명을 확인하고 제안 응답들을 대조하여 제안 응답이 동일한지 확인하는데 체인코드가 원장만을 쿼리한 경우, 어플리케이션은 쿼리 응답을 검사하고 ‘Ordering Service’에 Transaction을 보낼 수 없게 하였고, 클라이언트 어플리케이션이 지정된 보증 정책이 충족되는지 검사를 하고나서, 블록 내의 Transaction은 보증정책이 충족되는지 확인 작업과 Transaction 실행에 의해 원장 상태가 변경되지 않았는지 확인하기 위해 검증 작업을 할 수 있도록 실행한다.



그림 7. SKD 응답 점검
Fig. 7. SDK response check

3.3 검증

본 논문에서 제안한 Transaction을 검증하고자 스마트 팩토리의 디바이스 접근제어에 적용해 보고자 한다. 스마트 팩토리 내에는 활동성이 부여된 장비, 로봇, 생산 설비 등을 비롯하여 원자재, 완성품 등 센서 등이 부착되는데 여기서 다양한 엔터티들이 존재한다. 각 엔터티는 이벤트 업, 다운로드 등을 포함하여 다양한 접근제어의 관리 대상이 되고 있지만 현재는 효율성의 문제들로 인해 시리얼넘버 등의 단순 인증만을 사용하고 있다. 그래서 기존 검증연구 보다 이 논문에서 제안한 검증 효과를 보고자 Hyperledger Tool인 Composer를 설계하였다.

IV. 실험 결과

4.1 효율성 검증

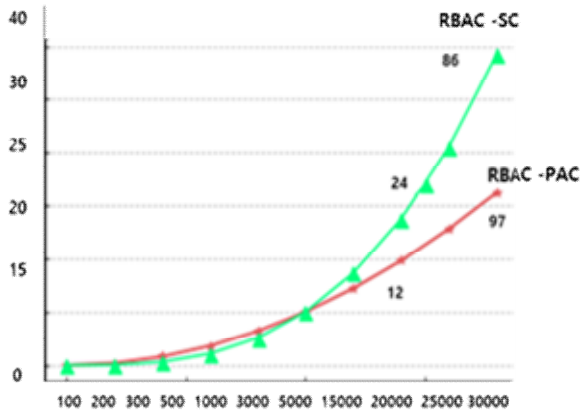
본 논문에서 제안하는 RBAC-PAC의 효율성을 비교하기 위해 필요한 Transaction을 Reg, Access Control, Privilege Renewal로 구분하여 조사하였으며, 총 Transaction 수는 표 1과 같다. RBAC-SC는 최소 6개 이상의 Transaction이 필요하다. 역할을 확인하기 위해 A.Control 세션당 Transaction을 수행하고 이러한 확인 절차를 디지털 서명으로 대체한다. 또한 RBAC-PAC은 단일 역할에 할당된 엔터티의 수를 30개로 가정하여 역할별로 갱신된다[12][13]. 또한 역할 갱신 시, Hyperledger Fabric에서는 역할 단위로 갱신하도록 가정하였다.

표 1을 기반으로 RBAC-SC와 Hyperledger Fabric의 효율성을 비교하기 위하여 첫 번째로는 하나의 엔터티가 등록, 접근제어, 권한갱신을 각 1회씩 했을 때 엔터티 수에 따른 효율성을 그림 8(a) 그래프에 나타내었다. 그림 8(b)는 하나의 엔터티가 1회 등록절차를 수행 후 10회 접근제어, 10회 권한 갱신했을 경우 효율성을 비교한 것이다[14][15]. (a) 그래프에서 엔터티의 수가 3만개일 경우에 41%로 줄어든 것을 볼 수 있다.

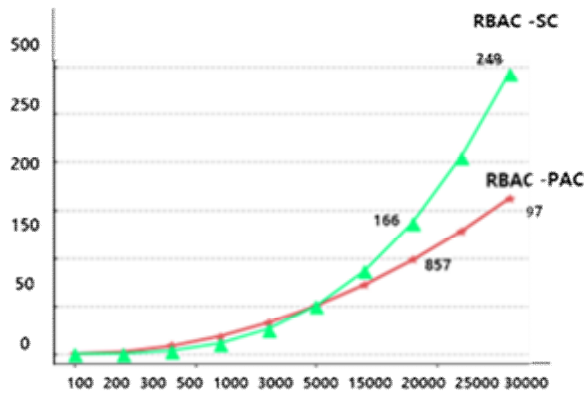
표 1. Hyperledger Fabric의 트랜잭션 비교

Table 1. Transaction comparison of hyperledger fabric

	N.Entities	Reg	A.Control	P.Renwal	TotalTran
RBA-SC	1	3	6	2	12
	300	900	1800	600	3600
Hyperledger fabric	1	2	3	1	7
	300	600	900	10	1810



(a) 엔터티 수에 따른 효율성
(a) Efficiency based on the number fo entities



(b) 엔터티 등록절차 따른 효율성
(b) Efficiency according to the entity registration procedure

그림 8. 비교
Fig. 8. Comparison

4.2 Realtime Processing

실시간 처리로 빅데이터 환경에서의 실시간 컴퓨팅을 엄격히 정해진 시간 내에 응답을 보장하고 즉 주어진 시간 안에 필요한 프로세싱을 해서 결과를 내주거나 처리를 하게 되면 실시간이라고 하는데 실시간 처리에는 다양한 영역들이 있다. In-Memory Computing, Interactive Processing, MapReduce,

Indexed Storage, Realtime Analytics가 있는데 본 연구에서는 Real Time Analytics를 연구하고 있으며 CEP(Complex Event Processing)이라고 불리는 다양한 실시간 이벤트를 분석할 수 있는 기술을 설계하였고, 데이터(이벤트) 모델링과 프로세스를 설계해서 적용하도록 구축하였으나 빅데이터에 적합한 수평적인 확장성(Scale-out)이 불가능하며 이벤트 스트림별로 여러 대의 서버로 부하 분산을 하거나 여러 개의 카드가 필요하며 메인 메모리를 갖춘 고성능 서버를 이용해야 하는 대량의 이벤트 처리가 가능하다. 그만큼 비용이 크며 중·소기업 스마트팩토리와는 맞지 않다.

그래서 본 논문에서 제안하는 실시간 처리 방법으로는 Real Time Anlytics의 Storm는 분산 실시간 프로세싱 구현의 복잡도를 낮춰주고 Clojure, Java, Python은 기본으로 제공하며 어떤 언어든 사용자가 익숙한 언어를 이용해 구현 할 수 있다. Node의 장애를 자동으로 관리해주고 Process, Server를 이용하여 병렬처리가 가능하도록 설계할 수 있고 추가 확장이 용이하다. Hadoop과는 달리 클러스터를 관리하는 작업이 매우 단순해 복잡한 설정이나 관리 포인트가 없어 테스트를 용이하게 할 수 있다.

4.3 Conclusion

본 논문에서는 스마트팩토리 내의 블록체인, 실시간 처리, Transaction 등 접근제어 방식을 정의하였다. 이 논문에서 효율적으로 사용하기 위해 계층적으로 설계하여 상속개념을 적용하였으며, 인증 및 허가 시에도 효율적으로 권한 속성을 확인하고 접근제어에 활용할 수 있도록 디지털 서명 후 블록체인을 이용하여 배포하였다. 이로 인해 반복해서 수행해야 할 될 등록 및 관리 작업 및 검증 작업을 단순화하여 효율을 높였으며, 이로 인해 효율성이 중요한 스마트팩토리 환경에서 새로이 배포되는 Hyperledger Fabric을 적용하여 활용할 수 있게 됨으로써 보다 안전하고 효율적인 접근제어가 가능하게 되었다. 향후, 보다 다양하고 효율적이면서 보안성이 강조된 블록체인 기법 등이 연구되어야 할 것이다.

References

- [1] Korea Smart Factory Foundation, "Case study of companies participating in Korea smart factory support project", Korea Smart Factory Foundation & Ministry of SMEs and Startups, Sep. 2017. <https://www.smartfactory.kr/datum/popup/datumDetail.do?dboardNo=165>. [accessed: Mar. 19, 2019]
- [2] Yuyu Yin, Fangzheng Yu, Yueshen Xu, Lifeng Yu, and Jinglong Mu, "Network Location-Aware Service Recommendation with Random Walk in Cyber-Physical Systems", *Sensors*, Vol. 17, No. 9, Sep. 2017. <https://doi.org/10.3390/s17092059>.
- [3] Yuyu Yin, Lu Chen, Yueshen Xu, Jian Wan, He Zhang, Zhida Mai, "QoS Prediction for Service Recommendation with Deep Feature Learning in Edge Computing Environment", *Mobile Networks and Applications*, Apr. 2019. <https://doi.org/10.1007/s11036-019-01241-7>.
- [4] Smart factory performance and Industry 4.0, Büchi, Giacomo; Cugno, Monica; Castagnoli, Rebecca Elsevier Science B.V., Amsterdam, 2020.
- [5] Gaige Chen, Pei Wang, Bo Feng, Yihui Li, and Dekun Liu, "The framework design of smart factory in discrete manufacturing industry based on cyber-physical system", *Taylor & Francis 2020 International journal of computer integrated manuf.*, Vol. 33, No. 1, pp. 1-23, Dec. 2019. <https://doi.org/10.1080/0951192x.2019.1699254>.
- [6] S. H. Hong. and H. J. Shin., "Analysis of the vulnerability of the IoT by the scenario", *Journal of the Korea Convergence Society*, Vol. 8, No. 9, pp. 1-7, Sep. 2017. <https://doi.org/10.15207/JKCS.2017.8.9.001>.
- [7] M. K. Kwak (Hanyang University), "Factory and smart factory", *Human & Digital Presentation material*, 2018.
- [8] I. S. Jeon, "Curriculum development for smart factory information security awareness training", *Journal of KIISC*, Vol. 26, No. 5, pp. 1335-1348, Oct. 2016. <https://doi.org/10.13089/JKIISC.2016.26.5.1335>.
- [9] S. H. Hong and S. H. Park, "The research on blockchain-based secure IoT authentication", *Journal of the Korea Convergence Society*, Vol. 18, No. 11, pp. 57-62, Nov. 2017. <https://doi.org/10.15207/JKCS.2017.8.11.057>.
- [10] Hong and C. R. Seo, "Developing a blockchain based accounting and tax information in the 4th industrial revolution", *Journal of the Korea Convergence Society*, Vol. 9, No. 3, pp. 45-51, 2018. <https://doi.org/10.15207/JKCS.2018.9.3.045>.
- [11] K. Blockchains, "Blockchains and smart contracts for the internet of things", *IEEE Access*, Vol. 4, pp. 2292-2303, May 2016. <https://doi.org/10.1109/ACCESS.2016.2566339>.
- [12] D. F. Ferraiolo, "Proposed NIST standard for role-based access control", *ACM Transactions on Info System Security*, Vol. 14, No. 3, pp. 224-274, Aug. 2001. <https://doi.org/10.1145/501978.501980>.
- [13] Oren Harell, "The Problem with Role Based Access Control", 30 May 2019. [Online]. Available: <https://blog.plainid.com/problem-with-rbac>. [accessed 11. 23. 2020].
- [14] YongJoo, Sang-Ho LEE, "Efficient RBAC based on Block Chain for Entities in Smart Factory", *Journal of the Korea Convergence Society*, vol.9, no.7, pp.69-75, 2018. <https://doi.org/10.15207/JKCS.2018.9.7.069>
- [15] A. J. LEE, "TOWARDS PRACTICAL AND SECURE DECENTRALIZED ATTRIBUTE-BASED AUTHORIZATION SYSTEMS", *University of Illinois at Urbana-Champaign*, Jul. 2008.

저자소개

남 기 훈 (Ki-Hun Nam)



2006년 2월 : 서경대학교

컴퓨터과학과(이학박사)

2006년 3월 ~ 2017년 2월 :

소프트웨어학과 겸임교수

2006년 8월 ~ 2009년 9월 :

한양대학교 디스플레이 연구소

연구원

2009년 10월 ~ 2011년 2월 : 충북대학교 BK21 연구원

및 초빙교수

2011년 3월 ~ 2017년 2월 : 서경대학교 컴퓨터공학과

초빙교수

2017년 3월 ~ 현재 : 서경대학교 컴퓨터공학과 조교수

관심분야: 인공지능, 사물인터넷(IoT), 임베디드 시스템