**Abstract**

IoT 장치의 보안 및 사생활 문제는 주된 이슈

경량화된 블록체인을 개발

이 논문에서는 스마트 홈 계층을 자세히 설명

**1. Introduction**

IoT 장치는 자원이 많이 없어서 따로 보안이나 개인정보를 보호할 남은 자원이 없다

기존의 보안 정책은 IoT 장치에 맞지 않다

IoT 장치는 분산된 보안을 원한다 -> 블록체인

기존의 블록체인을 적용하기에는 무리 -> POW 연산, 지연 시간 등등

그래서 경량 블록체인을 제안, 계층 구조를 가짐: smart home, overlay network, cloud storage

Smart home은 service provider, IoT 장치, cloud storage 등과 함께 overlay network를 구성

네트워크 오버헤드를 줄이기 위해 여러 노드를 클러스터로 묶고 각 클러스터에서 클러스터 헤더(CH)를 선택

CH들은 두 개의 키 목록과 함께 블록체인을 유지

이 클러스터에 연결된 스마트 하우스의 데이터에 접근할 수 있는 사용자의 PK 목록: requester

이 클러스터에 연결할 수 있는 스마트 하우스의 PK 목록: requestee

Cloud storage는 데이터를 저장하고 공유하기 위해 사용

이 논문에서는 smart home에 대한 세부사항을 논의

**2. Core Components**

**2.A Transactions**

로컬 장치나 overlay 노드 간의 통신을 트랜잭션이라고 한다

여러 종류의 트랜잭션이 생성됨

Ex) 데이터 저장을 위해 장치가 store transaction을 생성, SP나 집 주인이 Cloud storage에 접속하기 위해 access transaction을 생성, SP나 집 주인이 장치 정보를 주기적으로 모니터하기 위해 monitor transaction을 생성, 새로운 장치를 추가하기 위해 genesis transaction을 생성 등등

모든 트랜잭션은 공유키를 시용해 통신을 확보

트랜잭션 중 내용 변화를 검출하기 위해 lightweight hashing을 사용

Smart home과 관련된 모든 거래는 local private 블록체인에 저장

**2.B Local BC**

각 smart home에는 거래를 추적하고 사용자 정책을 시행하기 위한 정책 헤더를 가지고 있는 local private BC이 있다

Genesis transaction을 시작으로 모든 장치의 트랜잭션은 BC에 immutable ledger로 묶여 있다

각 블록은 블록 헤더, 정책 헤더(총 2개의 헤더), 트랜잭션을 포함하고 있다

블록 헤더 – BC의 immutable을 유지하기 위해 이전 블록의 hash를 가지고 있다

정책 헤더 – 사용자 정책을 시행하는데 사용

Local BC은 local miner에 의해 관리된다

**2.C Home Miner**

Smart home miner는 중앙에서 smart home에 관련된 트랜잭션을 처리하는 장치

Miner는 트랜잭션을 승인, 감시한다

Genesis transaction 생성, 키 배포 및 업데이트, 트랜잭션 구조 변경, 클러스터 형성 및 관리

모든 트랜잭션을 한 블록으로 모아 BC에 추가

추가 용량을 위해 local storage를 관리

**2.D Local Storage**

백업 드라이브와 같은 저장 장소

FIFO를 사용하여 데이터를 저장

**3. The BC-Based Smart Home**

**3.A Initialization**

IoT 장치 추가하는 방법

- Miner가 공유키를 사용해 genesis transaction을 생성

- Miner와 장치 사이의 공유된 키는 genesis transaction에 저장

정책 헤더 추가하는 방법

- 논문에 제시된 구조에 따라 정책을 만든 후, 첫 번째 블록에 정책 헤더를 추가

- Miner는 BC의 최신 블록에 있는 정책 헤더를 사용하므로 소유자가 정책 헤더를 최신으로 업데이트 해주어야 한다

**3.B Transaction Handling**

Smart 장치들끼리 직접 통신해야 하는 상황이 발생한다

Miner가 이를 위해 공유키를 장치들에게 분배

공유키를 받으면 장치들끼리 통신이 가능

장치가 로컬 저장소에 데이터를 저장하고 싶으면 공유키를 miner에게 받아야 한다

Miner는 장치에게 이 요청을 받으면 공유키를 만들어 장치와 로컬 저장소에 분배

공유키를 가지고 있으면 장치는 직접 로컬 저장소에 저장이 가능

장치들은 데이터를 클라우드 저장소에 저장하는 것을 요청할 수 있다.

클라우드 저장소에 데이터를 저장하기 위해서는 먼저 요청자는 블록번호와 해시를 포함하는 시작점이 필요

요청 받은 저장소는 시작점을 만들어 miner에게 전달

이후 기기가 클라우드 저장소에 데이터를 저장해야 할 때는 miner에게 데이터를 전송

Miner가 요청을 받으면 장치를 승인, 그 후 로컬 BC의 마지막 블록 번호와 해시 값을 추출해 store transaction을 생성 후 데이터와 함께 저장소로 전송

클라우드 저장소는 데이터를 저장 후 새로운 블록 번호를 miner에게 되돌려준다

이 외의 가능한 거래 – access transaction, monitor transaction

Access transaction은 miner가 overlay 노드들로부터 access transaction을 요청 받은 경우에 수행

요청받은 데이터가 로컬 저장소에 있는지 클라우드 저장소에 있는지 확인

로컬 저장소에 있으면 로컬 저장소에 데이터를 받아 요청자에게 전송

클라우드 저장소에 있으면 miner가 직접 데이터를 요청해서 받은 후 요청자에게 전송하거나 마지막 블록 번호와 해시 값을 요청자에게 전송

후자의 시나리오는 요청자에게 장치가 클라우드 저장소에 저장한 모든 데이터를 읽을 수 있는 권한을 준다

Monitor transaction에 의해 miner는 요청 받은 장치의 현재 데이터를 요청자에게 전송

일정 기간 동안 정기적으로 데이터를 받게 할 수도 있다

집 주인이 CCTV 등의 화면을 실시간으로 볼 수 있게 해준다

**3.C Shared Overlay**

한 명이 두 개 이상의 집을 가질 때, 집 주인은 miner와 로컬 저장소를 각각의 집에 따로 설치할 필요가 있다

관리 비용을 줄이기 위해 공유 overlay를 정의

공유 overlay는 공통의 miner에 의해 단일 주택처럼 중앙에서 관리되는 두 개 이상의 smart home으로 구성된다

공유 overlay가 단일 smart home과 다른 점 - 공유 BC 구조, 집과 miner의 통신

Initialization 단계에서 각 집은 디바이스에 대해 genesis transaction이 생성되고, 공유 miner가 이를 각 집의 genesis transaction(home’s genesis transaction) 으로 묶는다.

Miner와 같은 집에 있는 장치에는 변화가 없다

Miner와 다른 집에 있는 장치에는 데이터 전송을 위해 miner와 각 집의 게이트웨이 사이에 VPN을 설정해야 한다