Private/Consortium Blockchain Model을 이용한 질소산화물의 조작 방지 시스템 제안

국민대학교 정보보안암호수학과
이세윤 정서우 김은주 윤혜진 최지원

목차

출처 : 경향신문, 프레시안

http://news.khan.co.kr/kh_news/khan_art_view.html?artid=201904172125015&code=940100
http://www.pressian.com/news/article/?no=237237&utm_source=naver&utm_medium=search

1. 배경 및 목표

미세먼지 배출량 상습 조작한 LG화학·한화케미칼

2019.04.17일자 신문

"대기업들, 배출 미세먼지 수치 대거 조작"

정부 미세먼지 대책, '조작된 수치'가 근거?

대기오염물질 배출량을 상습적으로 조작한 기업들이 무더기로 적발됐다.

이들 대기업은 올해 초 미세먼지 줄이는데 자발적으로 동참하겠다며 환경부와 업무협약을 맺고도 측정치 조작 행위를 멈추지 않았다.

2015년 4월부터 최근까지 광주·전남 지역의 대기오염물질 측정 대행업체 4곳과 짜고 오염물질 배출농도를 낮게 조작했다. 환경부에 적발된 배출농도 조작 건수는 총 1만3096건에 달한다.

1. 배경 및 목표

기업들을 대상으로 사업장에서 발생하는
미세먼지의 원인인 질소산화물 관련 데이터들을
모든 곳에서 볼 수 있게 분산화 하여 저장을 목표

*먼지 데이터를 블록 체인으로 저장하고자 하는 이유

① 모든 네트워크 참여자에게 공개·보관·관리됨
② 특정 거래 정보를 조작이 힘듦

2. Blockchain

*Blockchain 종류

Public Blockchain	Private Blockchain	Consortium Blockchain

2. Blockchain – Private/Consortium

Private Blockchain

하나의 기관에서 독자적으로 사용하는 블록체인

탈중앙화 ↓

중앙기관의 의사결정에 따라 변경 가능

거래증명

중앙기관에 의하여 거래증명이 이뤄짐

모든 참여자가 데이터 검증
필요한 전력소모나 보상이 필요 없음
처리 속도 빠름

Consortium Blockchain

허가 받은 여러 기관들만 접근할 수 있는 블록체인

부분 중앙화

참여자들의 합의에 따라 변경 가능

거래증명

사전에 합의된 규칙에 따라 거래증명이 이뤄짐

3. 구현에 사용할 기기

Dust Sensor

SDS011

Raspberry Pi

Raspberry Pi 3

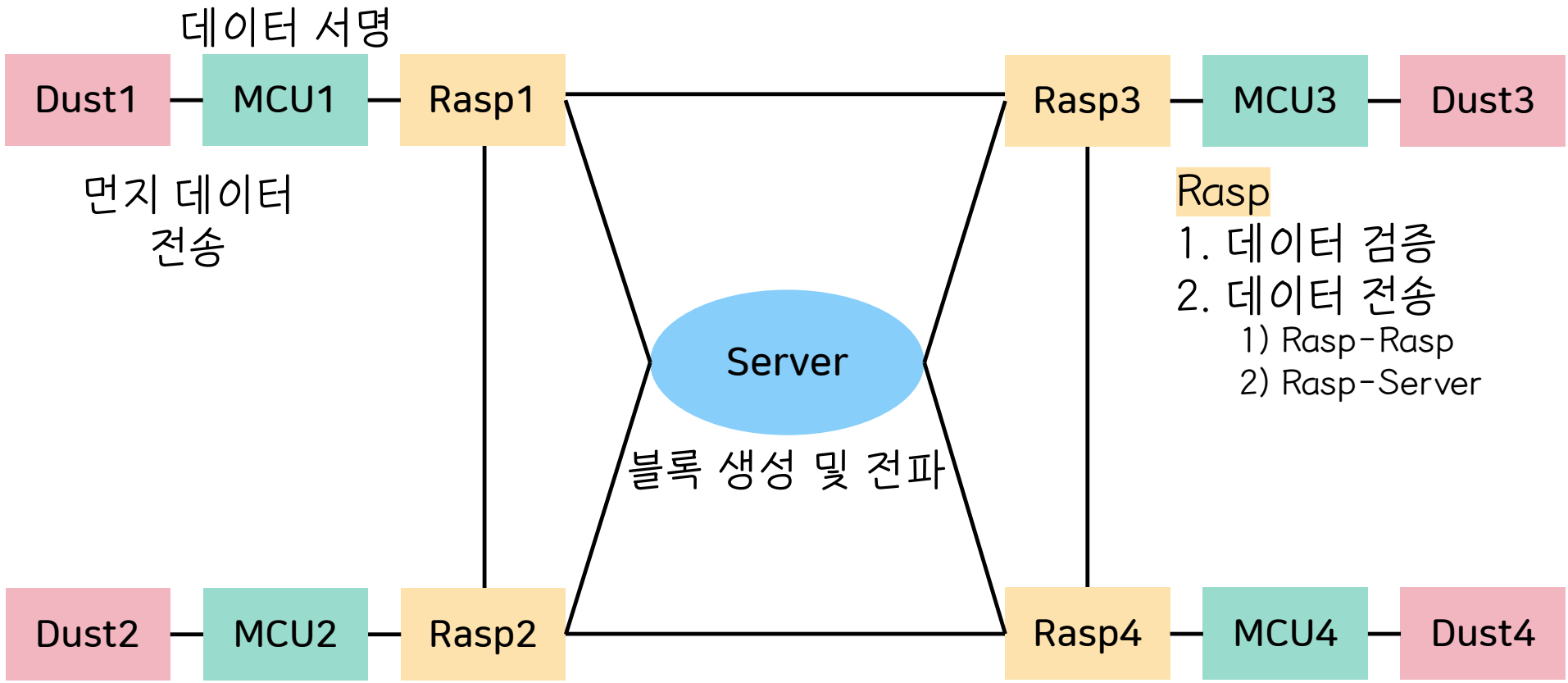
Debugger

SAM ICE

MCU

SAM G55 J19

4. 구성


데이터 서명
Dust1
MCU1
Rasp1
Rasp3
MCU3
Dust3
먼지 데이터
전송
Rasp
1. 데이터 검증
2. 데이터 전송
1) Rasp-Rasp
2) Rasp-Server
Server
블록 생성 및 전파
Dust2
MCU2
Rasp2
Rasp4
MCU4
Dust4

4. 구성

Dust — MCU

먼지 데이터 전송 -> 원본의 데이터를 전달
안전하다고 가정(물리적인 공격 X)

MCU — Rasp

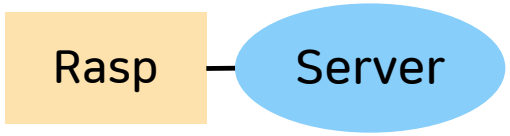
MCU에서 수집한 데이터를 MCU의 개인키로 암호화 한 뒤
Raspberry Pi로 전달 → ECDSA 사용

Rasp — Rasp

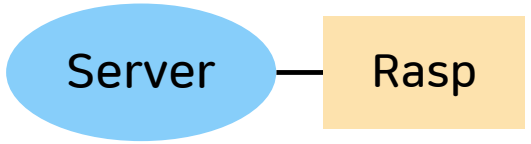
MCU로부터 받은 암호화된 데이터를 다른 Rasp에게 전달
데이터를 받은 Rasp는 자신과 연결된 다른 Rasp에게 전달 → TCP/IP 사용

각각의 Rasp는 암호화된 데이터를 검증 → MCU 공개키 사용

4. 구성



Server는 각각의 Rasp로부터 검증된 데이터 받기



51% 이상의 Rasp가 검증에 합의를 하면 블록 생성 후
데이터를 담아 다시 Rasp에게 전송
〉 Server는 무결한 존재라고 가정, 따라서 검증 필요 X

5. Protocol

Case ID

프로토콜	길이	ID

ID와 Server에 존재하는 ip.txt 안의 ID와 비교하여

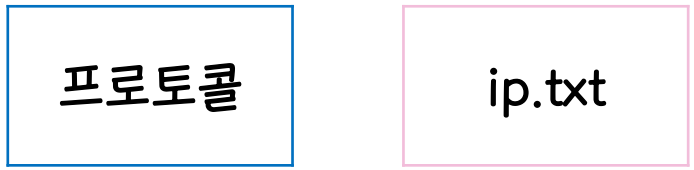
ip.txt에 존재하는 ID와
입력 받은 ID가 같고 IP가 같은 경우 ⇨ 종료 (0x03)

ip.txt에 존재하는 ID와
입력 받은 ID가 같고 IP가 다른 경우 ⇨ IP 갱신 (0x02)

ip.txt에 입력 받은 ID가
존재하지 않는 경우 ⇨ 등록 (0x01)

5. Protocol

Case IP



등록된 ip.txt 파일을 전송

Case DATA

입력한 ID의 txt 파일을 생성하여 받은 DATA를 등록
(파일이 이미 존재한다면 받은 DATA만 추가)

5. Protocol

Case VERI

프로토콜	길이	전파 ID	검증 ID	검증한 DATA	검증 결과

성공
(0x01)

전파한 ID에 해당하는 txt 파일에
검증 여부와 검증 결과를 등록한 뒤
검증 결과 부분이 전체 검증에 참여하는 Rasp의 과반수(51%)를
넘으면 검증된 Data로 보고,
txt 파일에 저장 (검증에 성공 시 1, 실패 시 0)

5. Protocol

Case BLOCK

프로토콜	block.txt

요청하면 만들어진 BLOCK을 전송

Case SEND

프로토콜	길이	ID	DATA	Time Stamp	서명

먼지 데이터를 받은 Rasp가
ip.txt을 확인하여 해당 Rasp에게 전송

6. Block 구성

Time Stamp	Number of Rasp.	Pre Hash	Size of Block	Number of Data	Data

- TimeStamp : 블록이 만들어진 시각
- Number of Raspberry Pi
  : 블록 데이터 생성 및 검증에 참여한 Rasp 개수
- Size of Block : 블록 크기
- Number of Data : 데이터 개수
- Data : 데이터

봐주셔서 감사합니다 ٨(ᐛ)٩