Distributed SUIT Architecture Model

Seoyun Choi,
Sangmyung University

Jong-Hyouk Lee,
Sangmyung University

Jung-Soo Park
ETRI
Contents

• Traditional SUIT Architecture

• Proposal

• Next Step
Traditional SUIT Architecture

- Adopting Client-Server model

- Manifests and firmware images are downloaded from ‘firmware servers’
Traditional SUIT Architecture

• Problems

1. Client-server architecture
   • can cause overhead on servers and update failures may occur
   • servers can be targeted by an attacker for use in an attack

2. Author-disappearing
   • If authors disappear, firmware consumers who have not yet updated to the latest version cannot catch up
Traditional SUIT Architecture

- **Author-disappearing issue**
  - Maintenance of servers is dependent on the author’s management
  - Data is not available without servers
    - e.g.,
Proposal

- Current SUIT architecture has shortcomings
  - adopting traditional client and server model
  - cannot deal with an ‘author-disappearing issue’

- Blockchain can solve the shortcomings
  - By providing distributed storage (database) for manifests and firmware image files
  - By providing irreversibility for manifests and firmware image files
Proposal

- Solving an Author-disappearing issue

- Even an author’s disappeared, data is keep stored on blockchain because it’s irreversible

  e.g.,

  Status of an Author
  Firmware Server
  Status Tracker
  Status of a Consumer

  firmware version 4.0
  firmware version 5.0

  Author disappearing

  success
  missing

  Update Completed
  Update Failed
Proposal

• Proposed architecture

• Firmware Server → Blockchain
  • Distributed storage
  • Data is irreversible

• provides
  • high availability
  • high reliability
Proposal

• Proposed architecture

• To resolve bottle-neck problem
  • = registration node
    • Process node registration based on IP
  • = retrieval node
    • Retrieve the ip, URL for downloading a firmware image
Proposal

- Private or Consortium platform by cases
  - For Large Companies producing IoT devices
    - Private Blockchain platform
  - SMEs with higher possibility of author-disappearing issues
    - Consortium Blockchain platform
Thank You!

• Next Step
  • Submit a information model draft and improve with comments and discussions
  • Join hackathon with implementation

• Contact Info
  • Speaker: Seoyun Choi
    • seoyun@pel.smuc.ac.kr
**Types of Blockchain Architectures**

- **Public vs Consortium vs Private**

  - **Public Blockchain**
    - permissionless
    - every node can read & write data
    - opened system to anyone
    - risky...

  - **Consortium Blockchain**
    - permissioned
    - selected nodes can read & write data
    - sharing system for an union of small companies

  - **Private Blockchain**
    - permissioned
    - selected nodes can read & write data
    - private system for a large company
TPS(Transaction per Second) and Confirmation

- **Bitcoin vs Ethereum vs Hyperledger Fabric**

**<Bitcoin>**
- Block interval: 10 minutes (600 seconds)
- Average number of transactions on a block: 4200
- \( \text{TPS} = \frac{4200}{600} \approx 7 \) (tps)
- Confirmation Time = 60 minutes

**<Ethereum>**
- Block interval: 12~15 seconds
- Average number of transactions on a block: 150~450
- \( \text{TPS} = \frac{150~450}{15} \approx 10~30 \) (tps)
- Confirmation Time = about 2 minutes (120 seconds)

**<Hyperledger Fabric >**
- Block interval: N/A
- Average number of transactions on a block: depends on customization
- \( \text{TPS} = \text{close to 3500 tps} \) (depends on customization)
- Confirmation Time = N/A