Blockchain Applications for IoT Automotive System

Alessandro Vizzarri, PhD

Department of Enterprise Engineering «M. Lucertini»
University of Rome Tor Vergata
alessandro.vizzarri@uniroma2.it

Outline

- IoT & Automotive
- Traditional Approach
- Blockchain-based Approach
- Blockchain Applications for IoT Automotive Systems
- Conclusions

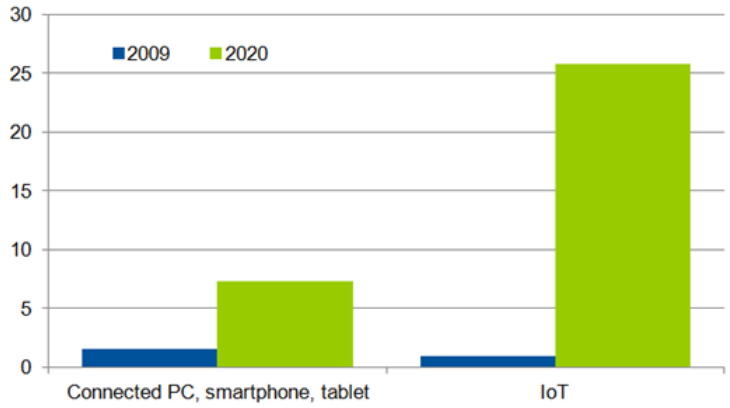
Outline

- IoT & Automotive
- Traditional Approach
- Blockchain-based Approach
- Blockchain Applications for IoT Automotive Systems
- Conclusions

IoT & Automotive

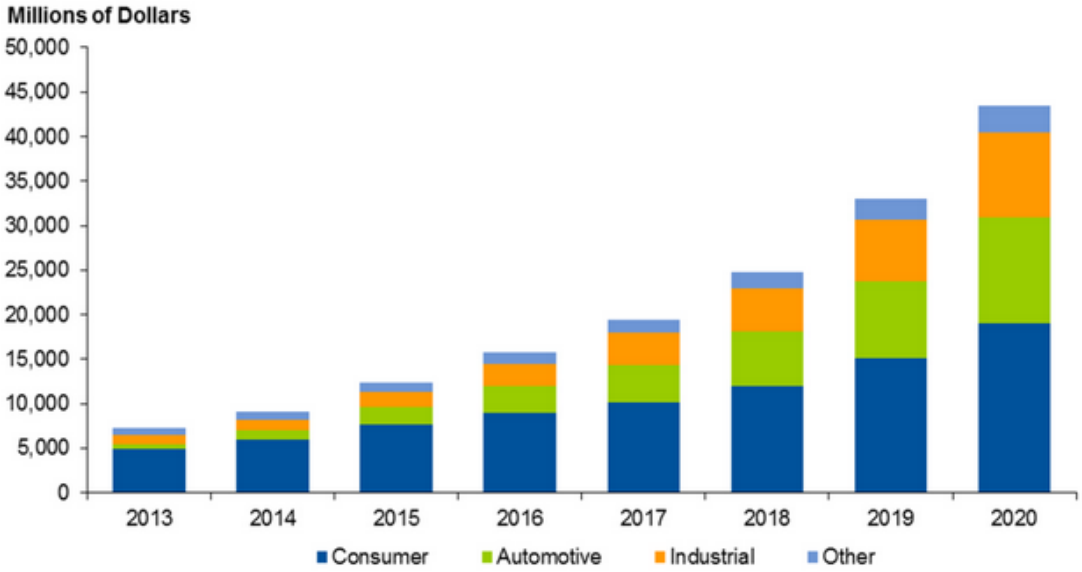
- Gartner predicts *"26 billion IoT units installed by 2020. IoT product and service suppliers are expected to generate incremental revenues exceeding $300 billion with a $1.9 trillion global economic added value."*
- IHS estimates that *"in 2019 about 5 billion of these devices will be business-critical devices."*

Total of Connected Deviecs, Billions of Units (Installed Base)



Source: Gartner (November 2013)

IoT Semiconductor Revenue by Electronic Equipment (Millions of Dollars)

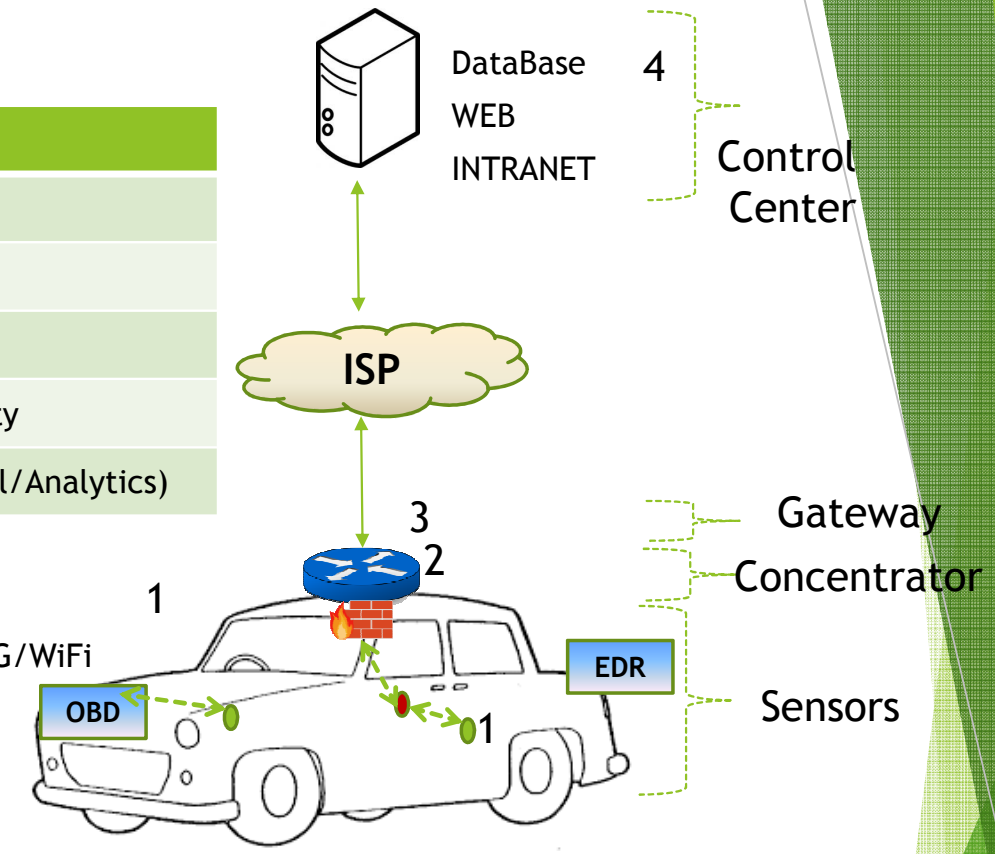


Outline

- IoT & Automotive
- **Traditional Approach**
- Blockchain-based Approach
- Blockchain Applications for IoT Automotive Systems
- Conclusions

Traditional Approach

IoT Subsystems for Automotive		
ID	**Name**	**Function**
1	Sensor/Actuator	Data Acquisition
2	Concentrator Node	Sensor MGT/Data Transfer
3	Gateway	Communication Connectivity
4	Control Center	Data Management (Archieve/Retrieval/Analytics)



- Data Acquisition from few sensors
- Brokered communication models
- **ONLY** Gateway Connectivity: VPN Tunnelling (PSK IKE & Ipsec) over 3G/4G/WiFi
- Sensor Application Protocol: HTTPS/ MQTT
- Centralized Data Management
- Necessity for Security (gateway & firewall)
- Trusted Third Party (TTP) for service & data certification (SSL/TLS)

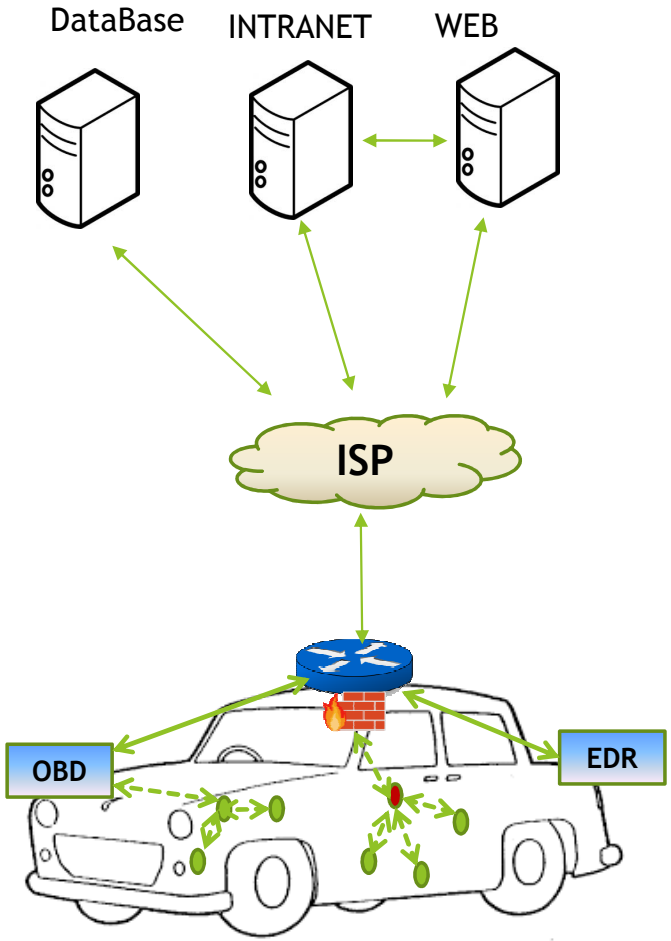
Security ISSUES

- Diagnostics equipment (On Board Diagnostic, OBD)
- VPN vulnerabilities (Port Fail & IP address display, Unauthorized Access, PSK IKE capturing, IKE Buffer Overflow, IPv6 leakage, ..)
- Identity management at different levels (vehicle, driver, sensor,..)
- Data can be corrupted or modified by attackers or fraudolent admins
- Data Integrity:
  - ✓Vehicle info: Event Data Recorder EDR, OBD, communications, localization,..
  - ✓Vehicle Insurance info: ownership, transfers, buy and sell...

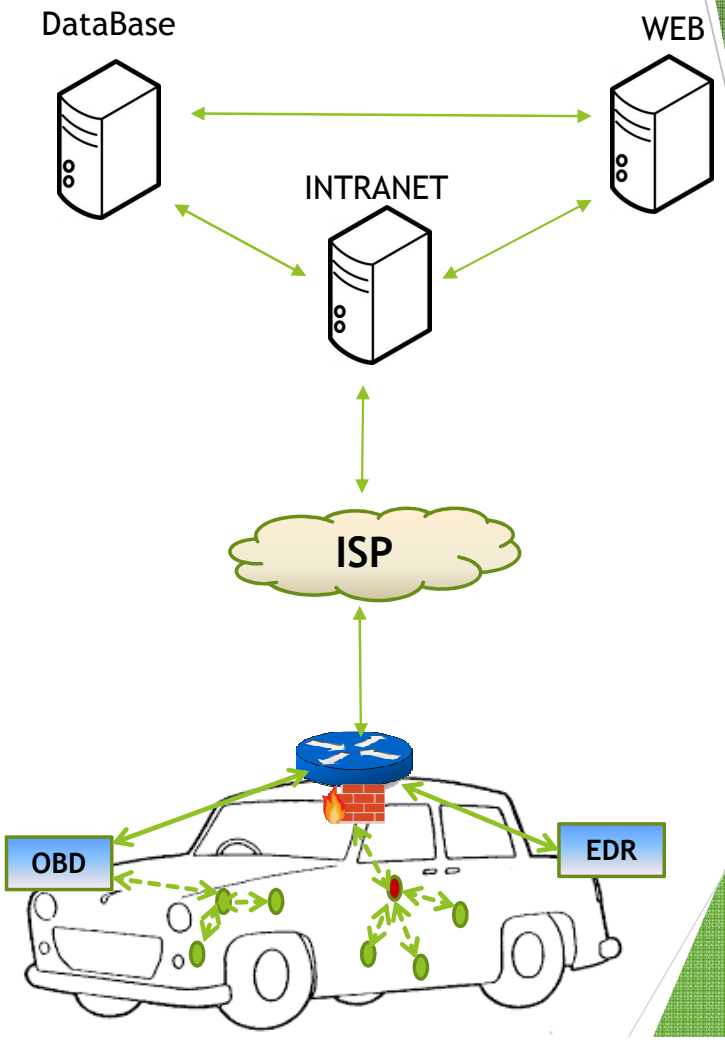
Last trends

- Decentralization of Data MGT
  - ✓ Impact of IoT
  - ✓ Edge computing (Fog Computing)
- Data Availability and integrity from Heterogenuos Data Sources → **Distributed Architectures**
- Data tracking (Time/Space)
- Network scalability
- Virtualization
- Necessity for interplatform communication → non homogeneous solutions

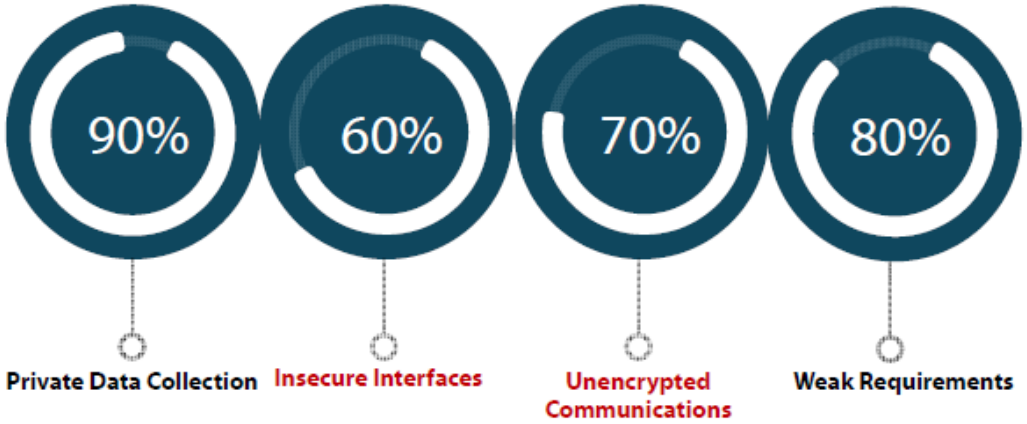
DECENTRALIZED



DISTRIBUTED



Common Security Incidents



Source: NIST

ID	Vulnerability	Issue
1	Insecure Web Interfaces	Default accounts, XSS, SQL injection
2	**Inefficient Authentication/Authorization**	**Weak passwords, no two-factor authentication**
3	Insecure Network Services	Ports open, use of UPnP, DoS attacks
4	**Lack of Transport Encryption**	**No use of TLS, misconfigured TLS, custom encryption**
5	Private Data	Unnecessary private information collected
6	Insecure Cloud Interfaces	Default accounts, no lockout
7	**Inefficient Mobile Interfaces**	**Weak passwords, no two-factor authentication**
8	Insufficient Security Configurability	Ports open, use of UPnP, DoS attacks
9	Insecure Software/Firmware	Old device firmware, unprotected device updates
10	**Poor Physical Security**	**Exposed USB ports, administrative accounts**

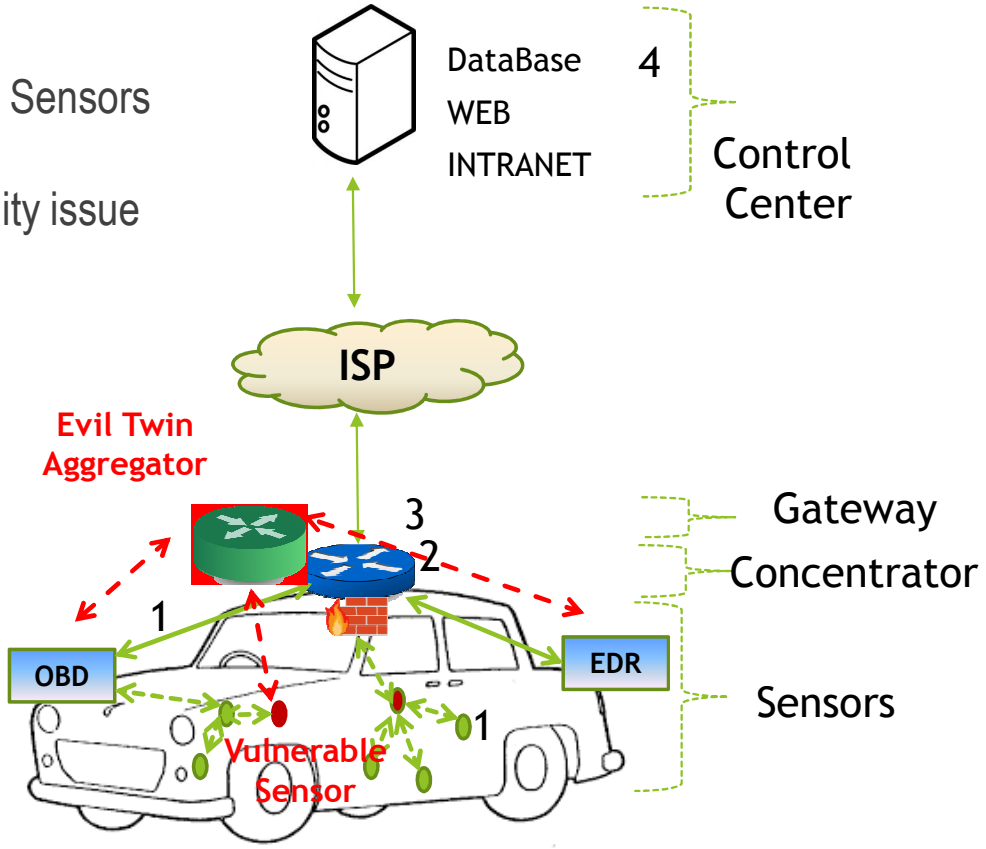
Source: OSWAP

Common Security Incidents

- Attacker introduces an Evil Twin Aggregator or Sensors with the same characteristics
- Vendor Dependent Custom Crypto: Compatibility issue
- No Authentication
- No Encryption

WHY?

- Wrong encryption managed by control center
- Custom TLS certification for speeding up the procedures → security holes
- Inefficient mobile interfaces (weak password)
- Physical Security: Exposed USB ports



IoT System Operational Requirements (DRAFT)

- Dynamic AND verifiable membership for a group of sensors / actuators
- Authentication & Data integrity
- Secure key leakage from a single-node perspective (or small sub-set of nodes)
- Encryption is a plus but not a firm requirement
- Sensor management with “sleep/power-off” periods
- Management of data from different sources

HOW?

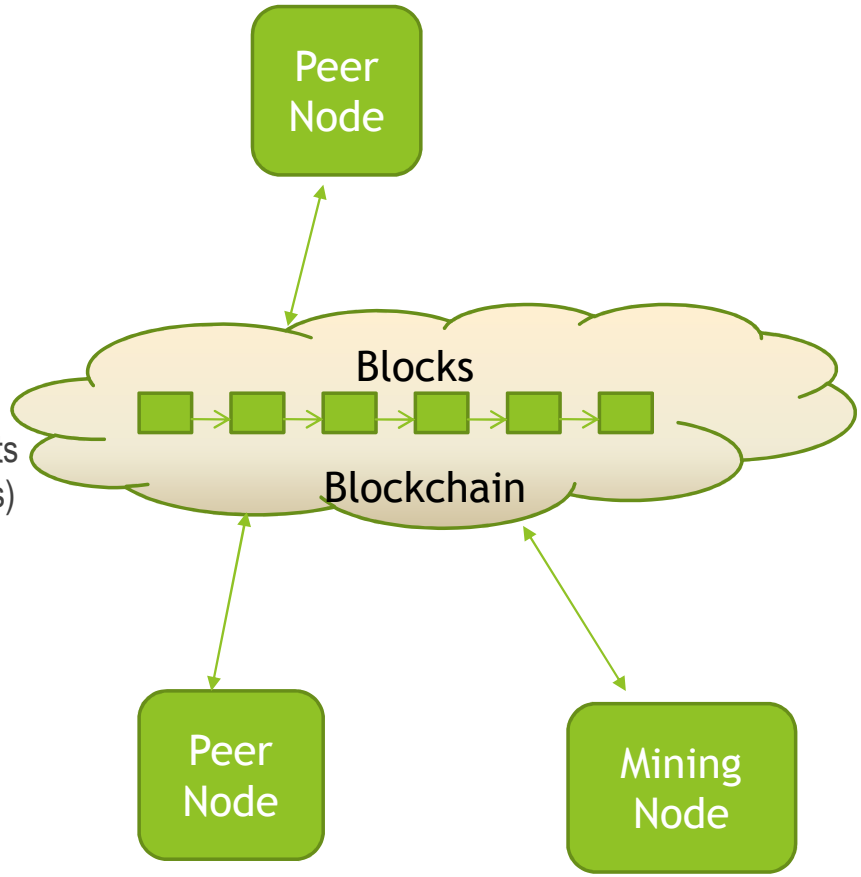
1. Lightweight Cryptography → In progress
2. Existing crypto blocks → microcontrollers supporting AES 256 cryptographic hashing → high costs
3. Blockchain → to be explored

Outline

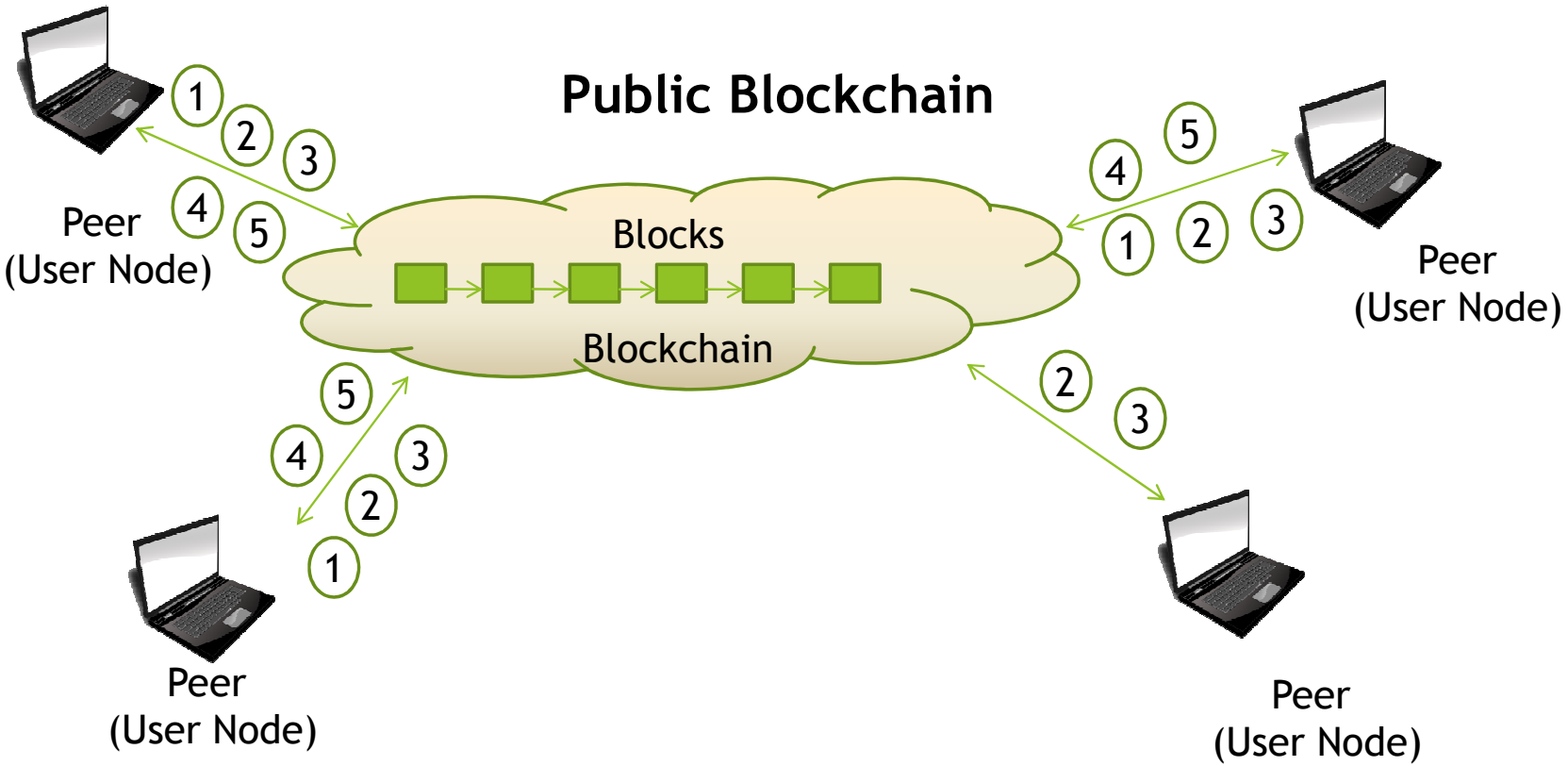
- IoT & Automotive
- Traditional Approach
- **Blockchain-based Approach**
- Blockchain Applications for IoT Automotive Systems
- Conclusions

Blockchain

- Blockchain refers to a “Public Distributed Verifiable Cryptographic Ledger”
  - **Public**: All participants gain access to “read”
  - **Distributed**: Peer-to-Peer Data Communication, Fully Decentralized
  - **Cryptographic**: Digitally signed transactions, proof-of-work limits rate of input (Asymmetric Cryptography: Public and Private Keys)
  - **Ledger**: Verifiable Transactional Database
- User’s nodes communicate in term of transactions
- The technology:
  - uses cryptography to authenticate and identify the nodes
  - allows them to securely add transactions to the ledger
- Transactions are verified and confirmed by other nodes (mining nodes) → no need for a central authority
- Blockchain can be **Public** (Permissionless) or **Private** (Permissioned)



Public Blockchain



1. Permission Assignment
2. Read
3. Write
4. Update
5. Delete

- Anyone can:
  - validate transactions (MINING)
  - add blocks
  - read data

Example of Public Blockchain: Bitcoin

Block Informations

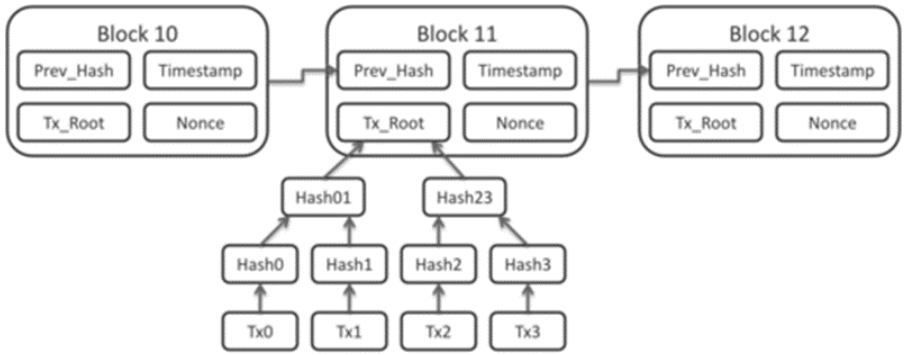
- Sequences of signed and verified transactions
- Published and distributed globally
- Magic number, Size
- Header
  - Hash of previous block (chain)
  - Merkle root hash of block
  - Timestamp
  - Target, nonce (mining)
- Number and list of transactions

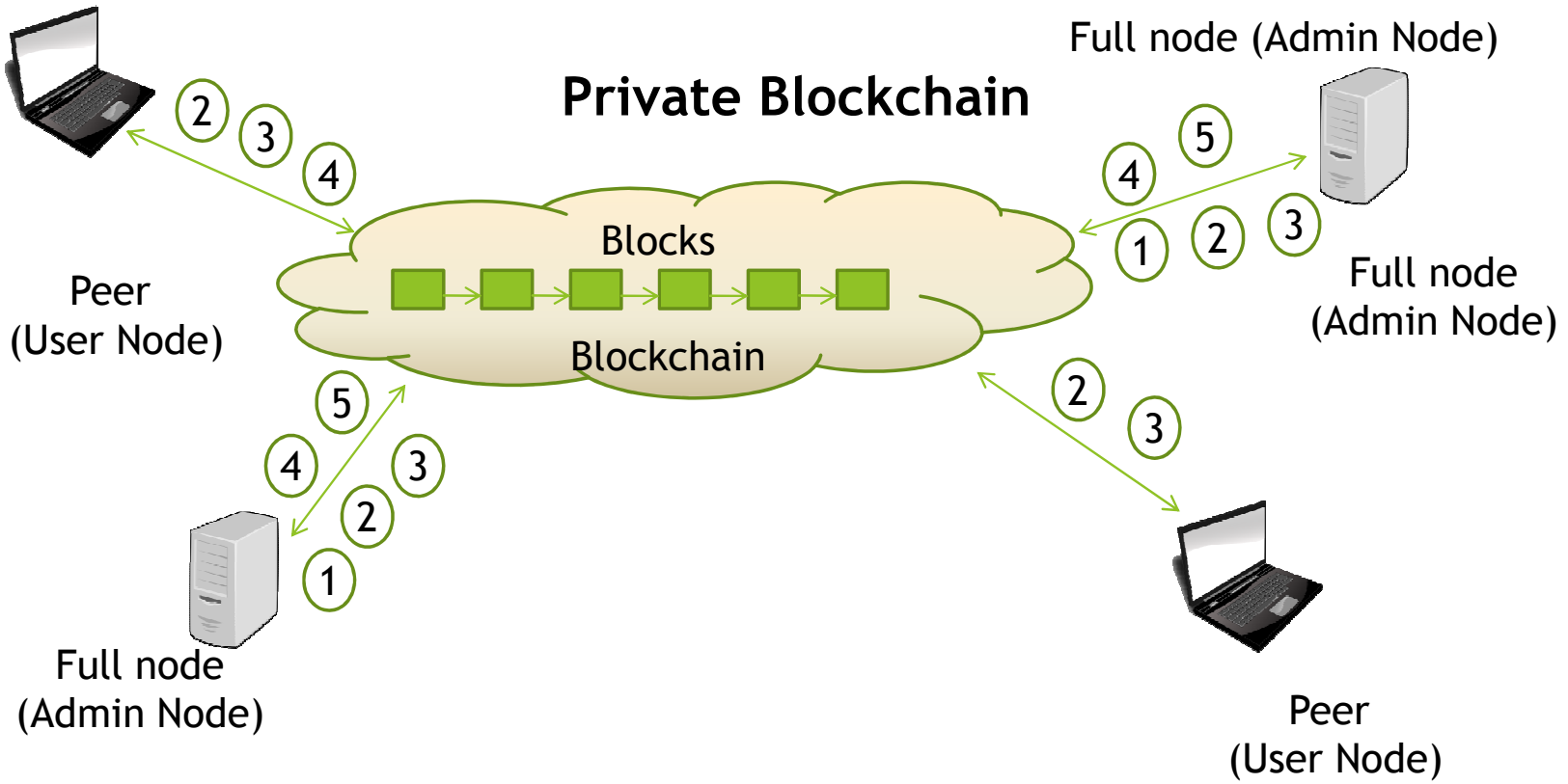
version	02000000
previous block hash (reversed)	17975b97c18ed1f7e255adf297599b55 330edab87803c817010000000000000000
Merkle root (reversed)	8a97295a2747b4f1a0b3948df3990344 c0e19fa6b2b92b3a19c8e6badc141787
timestamp	358b0553
bits	535f0119
nonce	48750833
transaction count	63
coinbase transaction	
transaction	
...	

Block hash

0000000000000000 e067a478024addfe cdc93628978aa52d 91fabd4292982a50

Source: S. Nakamoto





1. Permission Assignment
2. Read
3. Write
4. Update
5. Delete

- Only a limited number of entities (Fulll nodes) can validate transactions and add blocks
- They are known and allowed by the rest of the network
- They manage permissions for all nodes (peers)

Is the Blockchain suitable for IoT Systems?

- **Desiderable properties**
  - Distributed protocol with verifiable transaction history
  - Dynamic membership multi-party signatures

- **Undesiderable properties**
  1. Requires proof of “work”
  2. Requires Public Key Infrastructure (PKI)
  3. Size of the Ledger an issue for “small” devices
  4. Anonymous (not verifiable) Join/Leave operations

ACTIONS →

1. Requires **proof of earlier participation or proof of integrity** using history
2. Hash-based signatures (or other Merkle-tree schemes)
3. Prune and Compress Ledger. Maintain only transaction ledger for important devices
4. Group signatures using pre-shared group Key(s)

Private Blockchain can be the solution

IoT System Operational Requirements

- **Dynamic AND verifiable membership for a group of sensors / actuators**
- **Secure key leakage from a single-node perspective (or small sub-set of nodes)**

- Only Aggregators (FULL NODEs) can add nodes by issuing a group of Keys
- Can be done using Hash Chain
- Node is verified both by group key AND by participation history
- ***To add a node, an adversary will have to:***
  a) Compromise the group key
  b) Issue an "add node" transaction
  c) Add a sensor node
- Shape of the tree shows "additions" and "removals" of nodes over time

- **Authentication & Data integrity**

- Nodes and transactions are authenticated using the group key and the node Lamport signatures
- A node uses his Lamport public key to validate inserted DATA, transmits DATA to aggregator(s)

- **Encryption is a plus but not a firm requirement**

- No need for encryption

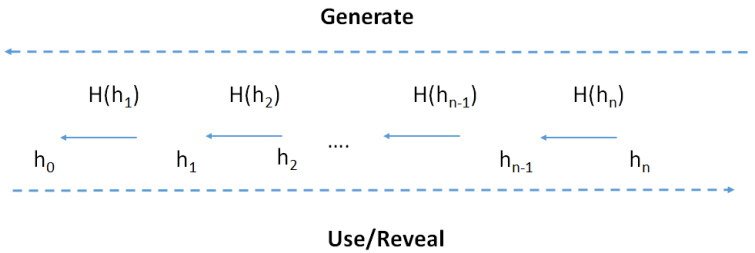
Private Blockchain can be the solution

IoT System Operational Requirements

Hash Chain

- **One-time hash password (Lamport,1981)**

- Chain is iteratively generated by applying a one-way hash function H
- s is a random seed
- Hash values are generated by iteratively hashing s (in reverse order of index):
  $H^i(s) = H(H^{(i-1)}(s))$, i=1,2,3,4,… s= "trust anchor"
- Hash chain includes a sequence of hash values (Public Keys):
  $h_1=H(s)$, $h_2=H(h_1)=H(H(s))$,…., $h_n=H(h_{n-1})$, n=1,2,…

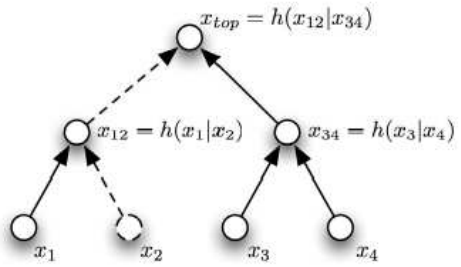


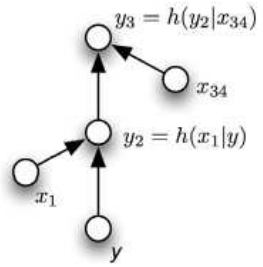
- h_0 is a commitment to the entire one-way chain
- any element of the chain is verified through h_0 (in the opposite order)

Example: Verify element s_i is indeed the element with index *i* of the hash chain
Solution: We have to check $H^i(h_i) = h_0$.
More generally: h_i commits to h_j if i<j
NOTES:

- to verify that h_j is part of the chain if we know that h_i is the *i-th* element of the chain, we have to check that $H^{j-i}(h_j)=h_i$
- We reveal the elements of the chain in this order h_0, h_1, …, h_{n-2}, h_{n-1} , h_n

Implementation





$$x_i = H\left(DATA||K_G||H^n(h_i)\right), H^{n-1}(h_i)$$

H=Hash
K_G= group Key
h_i= sensor *i-th* Public Key

Private Blockchain is the solution

IoT System Operational Requirements

Sensor management with “sleep/power-off” periods

- Nodes can be re-authenticated using their knowledge of historical transactions → they can prove their membership specific historical transactions using **predecessors** for Lamport Signatures

$$x_i = H\left(DATA||K_G||H^n(h_i)\right), k, H^{n-k}(h_i)$$

where *(n-k)* is smaller than the last signature from *i*

Management of data from different sources

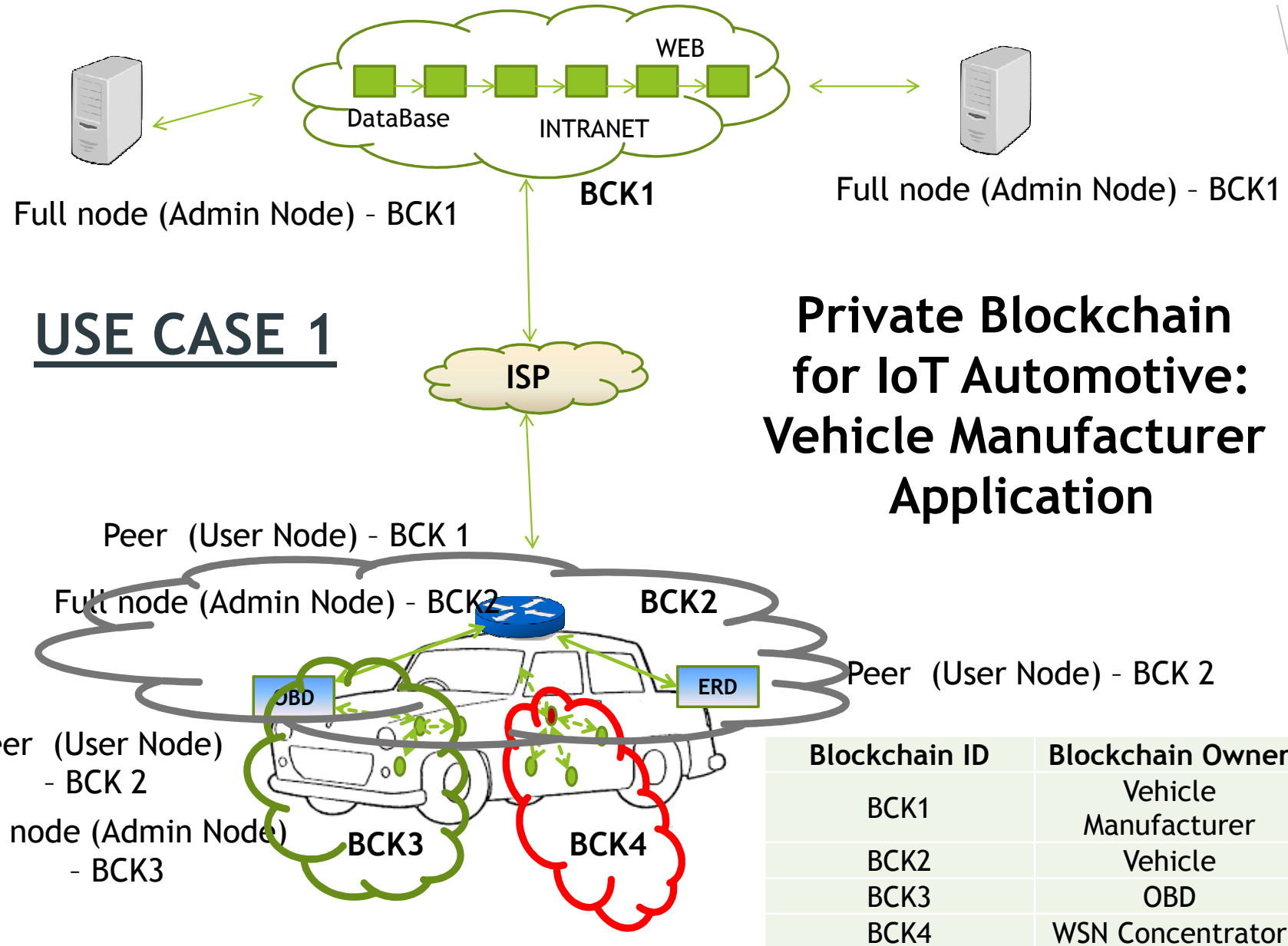
- Different nodes store different portions of the ledger
- Aggregators fully, others partial

Outline

- IoT & Automotive
- Traditional Approach
- Blockchain-based Approach
- **Blockchain Applications for IoT Automotive Systems**
- Conclusions

IoT Automotive Applications

- 3 Use cases related to IoT Automotive Sytems:
- Manufacturer
- Insurance
- eGov

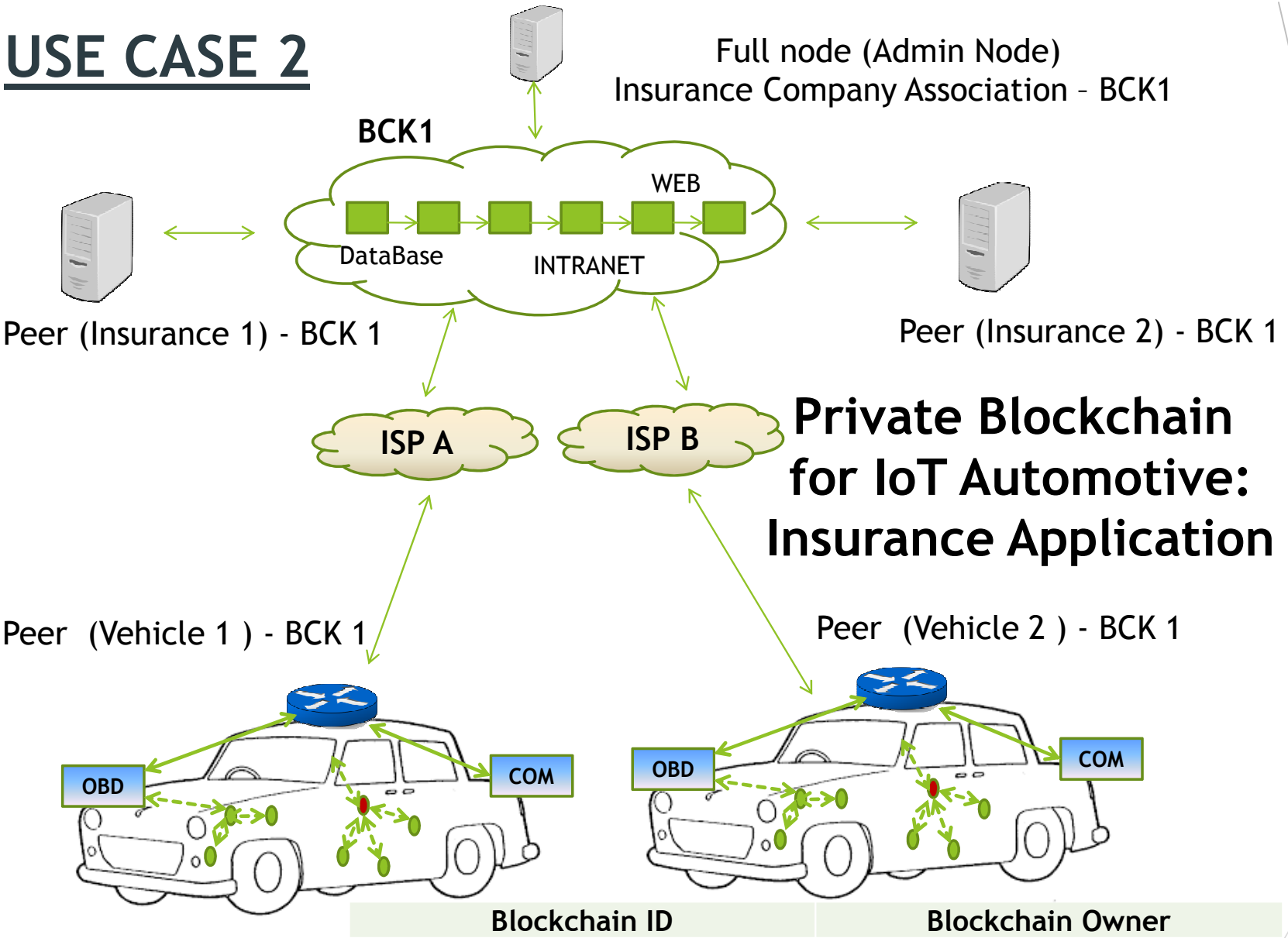


USE CASE 1

Private Blockchain for IoT Automotive: Vehicle Manufacturer Application

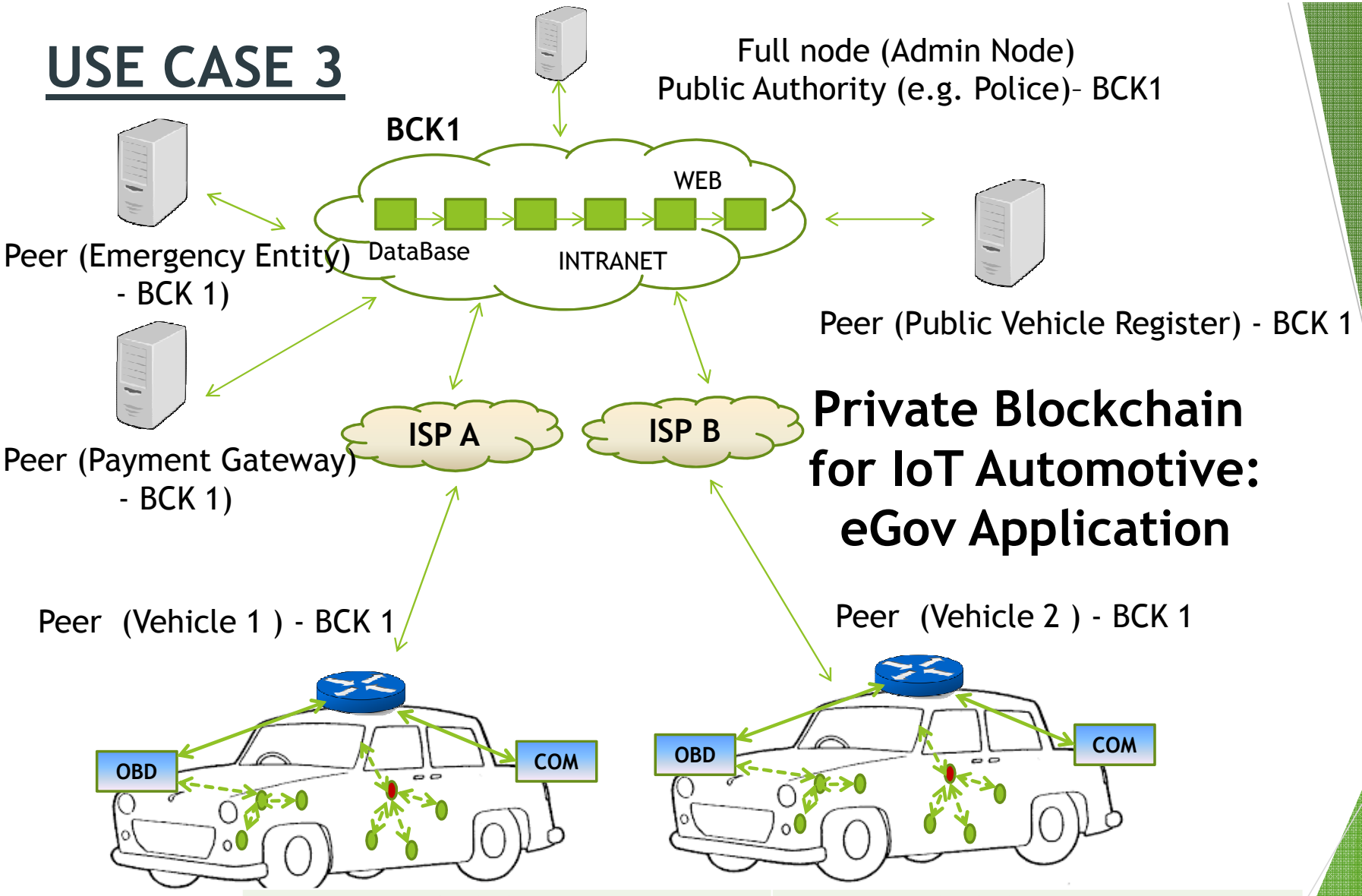
Blockchain ID	Blockchain Owner
BCK1	Vehicle Manufacturer
BCK2	Vehicle
BCK3	OBD
BCK4	WSN Concentrator

USE CASE 2



Blockchain ID	Blockchain Owner
BCK1	Insurance Company Association

USE CASE 3

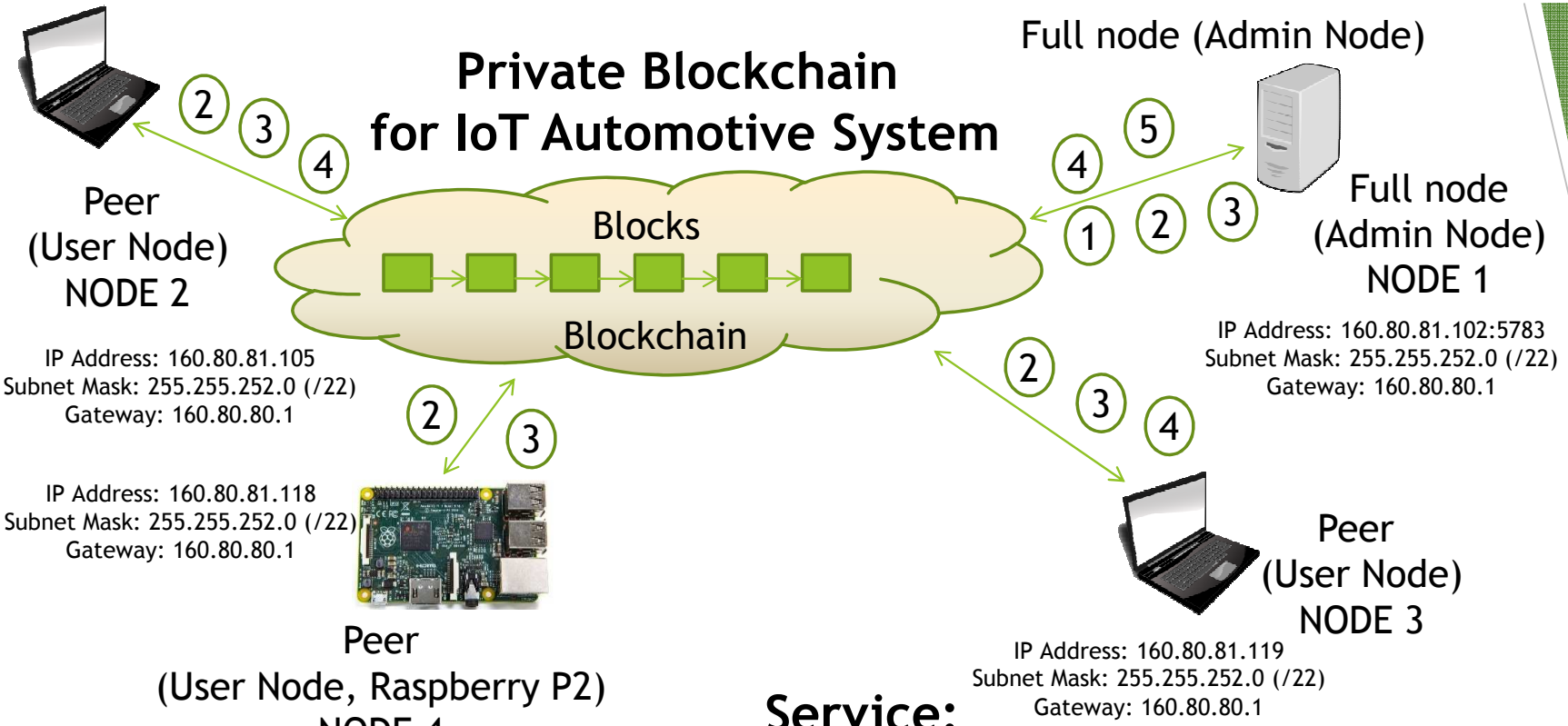


Blockchain ID	Blockchain Owner
BCK1	Public Authority (e.g. Police)

Implementations

- Test bed:
  - platform: Multichain 1.0 alfa-21
  - Operating System : Ubuntu 16.04 LTS
  - 4 devices (Nodes):
    - ✓ Node1: Full Node (ADMIN) → Fixed PC
    - ✓ Node2: Peer (User) → Laptop
    - ✓ Node3: Peer (User) → Fixed PC
    - ✓ Node 4: Raspberry P2 → (Ubuntu 14.04, WiFi Connection)

Private Blockchain for IoT Automotive System



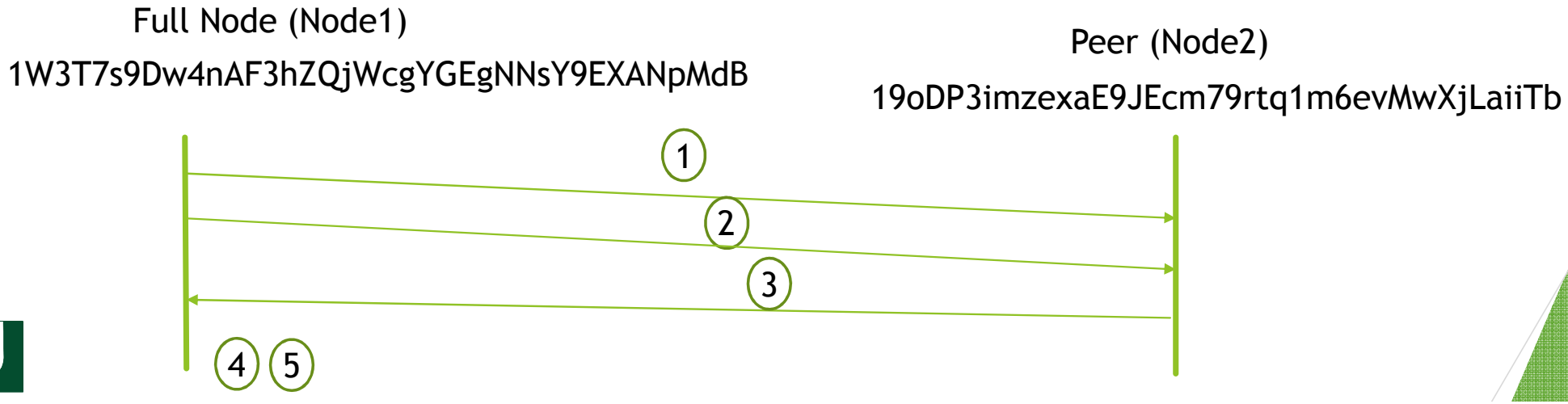
1. Permission Assignment
2. Read
3. Write
4. Update
5. Delete

Service: Data Streams Exchange

- All nodes send data streams to The Full Node (Node1)
- Functions based on OP_RETURN (Bitcoin)
- Data streams:
  - ✓ dimension: 4 KB
  - ✓ Hashed: SHA(256)
  - ✓ Content: Log files, Passwords, URLs for data repository

Examples: Chain1 (Node2 → Node1)

1. Node1:granting the connection of Node2
2. Node1: subscribing to a Data Stream (from Node2)
3. Node2 sending the Data Stream to Node1
4. Node1 retrieving the Data Stream from Node1
5. Node1 confirming the block (Mining)



Chain1: Block Parameters

After the Node1 mined the First Block (Block 1) of Chain1
Note: Block 0 is the Genesys Block by default

Block Parameter	Value
confirmations	37
Block hash	0000e824c2c00366481e7154000fe31ebc84eccd67b9e8c354aeb7795e380801
Block index	1
blocktime	1466675463
Tx id	e6e5e92221a051095f44bed2b50c81a50bd34b96345fbc7af5034d0f889e8e34
Time received	1466675459

Example (Node2 → Node1)

1. Node1 grants the connection of Node2

grant 19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb receive

{"method":"grant","params":["19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb","receive"],"id":1,"chain_name":"chain1"}

TxID: e6e5e92221a051095f44bed2b50c81a50bd34b96345fbc7af5034d0f889e8e34

2. Node1 subscribes to a Data Stream (from Node2)

importaddress 19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb

{"method":"importaddress","params":["19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb"],"id":1,"chain_name":"chain1"}

3. Node2 sends the Data Stream «0123456789abcdef»

chain1: sendwithmetadata 19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb 0 0123456789abcdef

{"method":"sendwithmetadata","params":["19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb",0,"0123456789abcdef"],"id":1,"chain_name":"chain1"}

Tx_ID: 2a43e9dbb0426f75038e87560f2993d83f69fbeb18aaf8acc0668ea250e7b92c

Example (Node2 → Node1)

4. Node1 retrieves the Data Stream from Node1

chain1: listaddresstransactions 19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb 5

{"method":"listaddresstransactions","params":["19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb",5],"id":1,"chain_name":"chain1"}

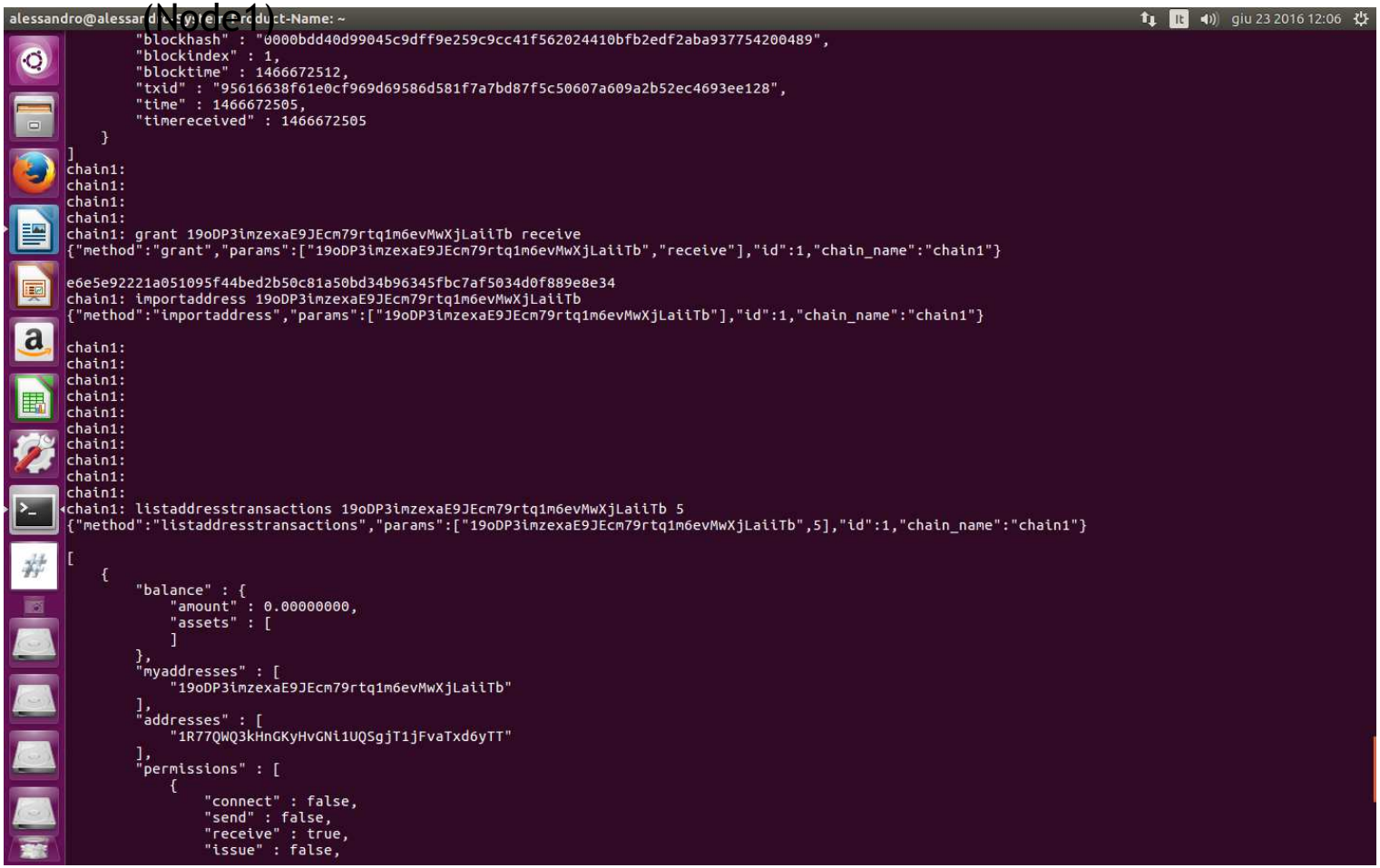

```
[
  {
    "balance" : {
      "amount" : 0.00000000,
      "assets" : [
      ]
    },
    "myaddresses" : [
      "19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb"
    ],
    "addresses" : [
      "1R77QWQ3kHnGKyHvGNi1UQSgjT1jFvaTxd6yTT"
    ],
    "permissions" : [
      {
        "connect" : false,
        "send" : false,
        "receive" : true,
        "issue" : false,
        "mine" : false,
        "admin" : false,
        "activate" : false,
        "startblock" : 0,
        "endblock" : 4294967295,
        "timestamp" : 1466675459,
        "addresses" : [
          "19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb"
        ]
      }
    ],
    "data" : [
    ],
    "confirmations" : 37,
    "blockhash" : "0000e824c2c00366481e7154000fe31ebc84eccd67b9e8c354aeb7795
    "blockindex" : 1,
    "blocktime" : 1466675463,
    "txid" : "e6e5e92221a051095f44bed2b50c81a50bd34b96345fbc7af5034d0f889e8e3
    "time" : 1466675459,
    "timereceived" : 1466675459
  },
  {
    "balance" : {
      "amount" : 0.00000000,
      "assets" : [
      ]
    },
```



```
    "myaddresses" : [
      "19oDP3imzexaE9JEcm79rtq1m6evMwXjLaiiTb"
    ],
    "addresses" : [
      "1W3T7s9Dw4nAF3hZQjWcgYGEgNNsY9EXANpMdB"
    ],
    "permissions" : [
    ],
    "data" : [
      "0123456789abcdef"
    ],
    "confirmations" : 19,
    "blockhash" : "00007ebc55fd58c41bc02f4719c3225afc925bbf8be331594fda8be63e3d047e",
    "blockindex" : 1,
    "blocktime" : 1466675690,
    "txid" : "2a43e9dbb0426f75038e87560f2993d83f69fbeb18aaf8acc0668ea250e7b92c",
    "time" : 1466675689,
    "timereceived" : 1466675689
  }
]
```

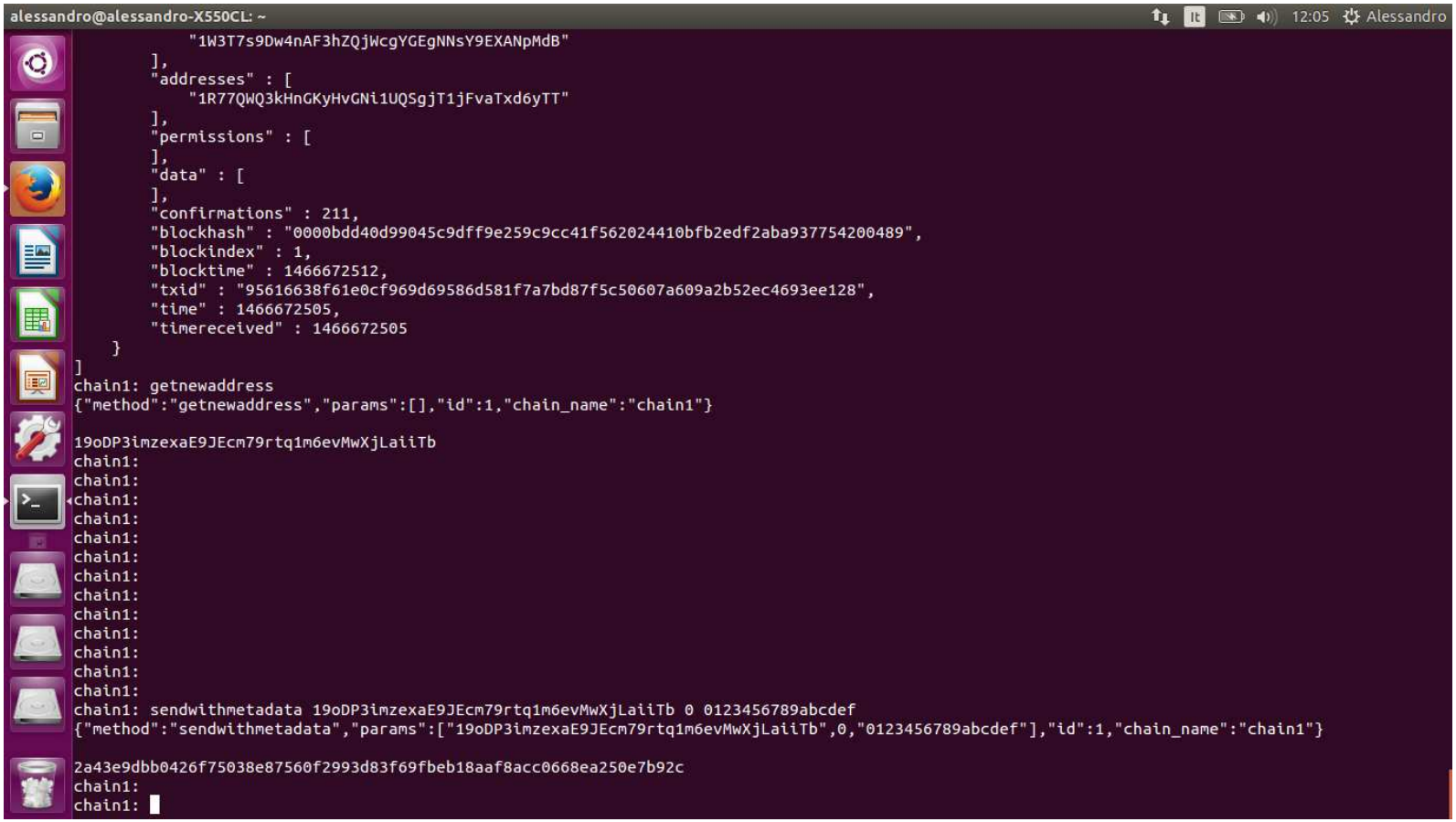
Screenshots (Node2 → Node1)

Full Node (Node1)



Screenshots (Node2 → Node1)

Peer (Node2)



Future work

- Implementation of Digital Signatures for each peer combined with SHA(512) e MD5
- Different Hash Chain Algorithms

Remarks

- VPN tunnels NOT involved
- TTP NOT involved
- Peers and Full Nodes:
  - are identified by a Public Address or Public Key [e.g. 25-byte binary address]
  - are equipped by a public and private keys
- Full Nodes [Admin] only manage peer permissions [send/receive/update/delete]
- Full nodes assign addresses and permissions to the peers
- Data Blocks are concatenated:
  - The timestamps have progressive values (not corruptable by peers)
  - The Data are traceable
- Data are only exchanged among peers belonging to the same blockchain

- Disintermediation
- Distributed architecture → distribution of consensus
- Data exchanged with digital signatures
- Data, event and time traceability
- Trustless
- Multiple transactions (not necessary monetary)
- Usage of powerful hash algorithms [SHA(256), SHA(512)]

Outline

- IoT & Automotive
- Traditional Approach
- Blockchain-based Approach
- Blockchain Applications for IoT Automotive Systems
- **Conclusions**

Conclusions

- IoT Scale, Vendors, Technologies are growing up very rapidly
- IoT Devices will always have diverse capabilities & Resources
- Use of Cryptography is done without clear understanding of the implications
- No Current Standards for Lightweight cryptography
- Blockchain inspired protocols combined with new cryptographic primitives might be the useful solution

Thank You !

Alessandro Vizzarri, PhD

Department of Enterprise Engineering «M. Lucertini»
University of Rome Tor Vergata
alessandro.vizzarri@uniroma2.it