Blockchain Mechanisms for IoT Security and Privacy

Huipeng Zeng

***Abstract*—Due to the growth and development of Internet of Things(IoT), we need measures to ensure the security and privacy of IoT. Since the emergence of blockchain technology and its high security, it has been widely implemented to enhance the security of IoT. From Network respect, we found it works well in decentralized access control with a blockchain mechanism inside. This paper highlights the advantage of decentralized access control and shows an access control framework adapted with blockchain technology. It further presents a case study of smart home with a lightweight blockchain embedded for access control..**

I. Introduction

With the explosive growth of electronic products in high technology society, the demand for an efficient managing manner for headless devices has become urgent. Internet of Things (IoT) weaves all the resources, that human can and are allowed to control so far, into a web, where shared data can be accessed and the resources can be managed; contributing to the sustainable development of the world. The biggest difference between Internet and IoT is that the Internet lets human acquire, exchange and save data. IoT makes machines do this on the background of current human-made rules. The current agenda of many researchers is to investigate how to create or improve rules to achieve a higher level of efficiency.

At present, the application of IoT in real life has particularly the following popular directions: Smart homes, smart cities, smart transportation, connected health, and industrial internet. The investments of enterprises and industries in IoT help to bridge the gaps between fields like finance, banking, marketing, create novel models of inter- and intra- organizational structures and enhance the efficiency of all activities. The various applications of IoT in personal and social domain then enable to connect individual users and the society into a common interconnected web which facilitates mutual interactions between humans and the surrounding environment.

In the first session, basic context and knowledge of IoT and blockchain technology are introduced. In particular, static description of three layers of this technology and security and security challenges of each layer. The focus of the next section is then the Network layer, introduction to authentication as well asa discussion of the advantages and disadvantages of centralized and decentralized access control. After that, we will see how the blockchain mechanisms be used in decentralized access control by the framework – FairAccess. Last but not least, a case study of a similar framework that can be used to implement a smart home system is presented.

A. Architecture of IoT

IoT is a system composed of three general groups, or layers, which are mutually interconnected. Each of the layers serves a specific purpose.

1) Perception Layer: Perception layer is also known as ”Sensors Layer”. This is so, as the sensors present here, sense some physical information like parameters or identifies of other smart objects in the environment and translate the perceptions into data. These are then passed to the upper layer, the network layer.

2) Network Layer: The Network Layer serves data transmission between different IoT devices by usage of modern technologies such as WiFi, LTE, 4G or Zigbee. The information generated by different sensors from the Perception Layer is processed and transmitted to Application Layer.

3) Application Layer: The Application Layer delivers application-specific services to the user with the guarantees of authenticity, integrity, and confidentiality of the data, and founds a smart environment like smart homes, smart health, and smart cities.

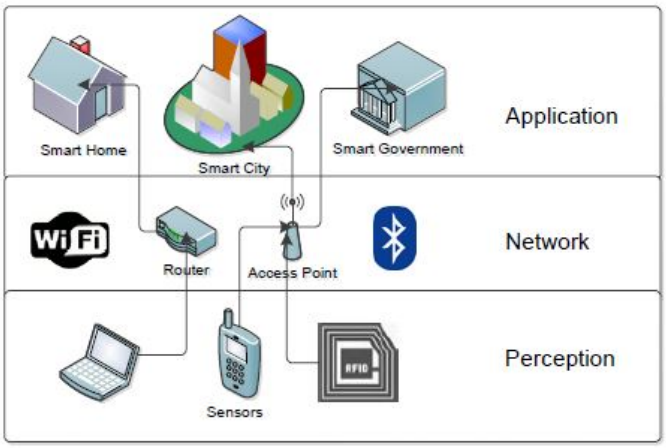

Fig. 1. Structure of Three Layers [10]

B. Security in IoT

1) What is Security in IoT: In general terms, IoT security means the protection against a third-party infringement. It’s the state or the quality of being secure. The objective of security is to ensure that only authorized users have access to the service and that unauthorized persons cannot obtain information or have a chance to falsify any information. A

successful organization should contain the following security levels in a system. [5]:

- Physical security: the protection of physical objects, items, or areas from unauthorized access
- Personnel security: the protection of individual or groups of individuals who can formally access the organization and its operations
- Operations security : the protection of pieces of data that generated from a particular operation or a serials of actions.
- Communications security : the protection of via telecommunications transmitted data and equipment that involves in the transmission.
- Network security: the protection of networking components, connections and contents.
- Information security: the protection of confidentiality, integrity and availability (known as C.I.A) of information assets.

2) Security challenges in different Layers:

- Perception Layer
  The challenges to security in perception layer originate mostly from sensors technologies like Radio-Frequency Identification (RFID), Smart card, Sensor network etc. These are the leading causes forsensor attacks, sensor abnormalities, radio interference.
  At the perception layer, a large number of sensors is needed for real-time data collection and fast and seamless delivery to the user. The authentication and data integration enter the process in this layer as well. Due to the wireless nature of the ongoing communication, leakage of confidential information, tampering, terminal virus and, copying etc are amongst the most profound security threats.
  Attacks often occur during data transmission, acquisition, integration, and processing. These Include: *Unauthorized access to the Tags* as a consequence of the lack of proper authentication mechanism in lots of RFID systems. *Tag Cloning* causes that the reader cannot distinguish between the original and the compromised tag. *Spoofing* gets a full control over the system, thus making it severely vulnerable.
  As a result, the perception layer needs four have basic elements of security which include Authentication, Data Privacy, Privacy of sensitive information and Risk Assessment.
- Network Layer
  Since a network servers for transmission of information, it has to ensure that data are transmitted securely. The security threats to the network layer can be grouped in two categories: security risks of the IoT, and risks relating to technologies and protocol defects during the design and implementation of a network [15]. The issue is that freely moving nodes in a wireless networks can casually join or leave the network at any time without any prior authentication. As a result, the network is vulnerable to malicious elements.. A solution to this problem is one the most active areas of research these days.
  Apart from DoS attacks, the adversary can also attack the confidentiality and privacy at network layer by traffic analysis, eavesdropping, and passive monitoring [2].Some of the related issues include the following. *Denial of Service (DoS) Attack*, is a kind of an attack where the network is flooded by a lot of unnecessary traffic by an attacker. This results in a resource exhaustion of the targeted system due to which the network becomes unavailable to the users [14]. *Man-in-the-Middle Attack* (MITM) is an attack where the attacker secretly relays and possibly alters the communication between two parties who believe they are directly communicating with each other. The unauthorized party can also possibly falsify the identity of the victim and communicate in a regular way to gain more information [13]. MITM can be also followed by eavesdropped attack.
  All these threats and risks, including illegal access to the sensor nodes to spread fake information, can be reduced or even prevented with the help of a proper authentication process and point to point encryption.
- Application Layer
  Several issues arise as a consequence of a lack of global policies and standards, controlling the interaction and development of applications. Different authentication mechanisms in different applications lead to the difficulty of protecting data privacy and issues with identity authentication. Another challenge is the design of IoT user management. Different users have different roles in the system. Users should in principle know how to manage the parts of the systems, how to transmit data and how to manage the data flows between users and over time. In this respect, the main security challenge are the users themselves; any potential flaw in the design can attract potential attacker. [4].

C. Blockchain Technology

1) Key terms in Blockchain: Before we discuss what blockchain is, several key terms in blockchain technology are explained at first.

- Hash Function and puzzle problem:
  Hash function refers to a method that converts the arbitrary size of data into a digital string of predefined fixed length, which is called hash or hash value. The hash value is representative of the original string or characters but is normally smaller than the original. Any digital string can be calculated as a large number; this number can then be divided by a constant with the remainder being then the result, hash. Although this kind of hash function is simple, it is used rarely, as we require a so-called one-way-computation, which makes back computing difficult. Hash function with such a property is called cryptographic hash function. It digests the bits of the input data in a very convoluted way to make the reversible computation impossible. The MD5, SHA1, SHA2 (SHA256 included). And in blockchain technology, we want to avoid attacks and control the speed of mining. For that, it is necessary to form protocols, we use a client puzzle as

a cryptographic defence against the attacks, which makes the back computing harder than usual.

- Hashchain and blockchain:
  Hashchain is a sequence of homogeneous data chunks, or simply blocks, linked together by a hash function, and consists of the hash and the payload [11].
  The process to generate a new hash block demands the participation of previous hash block. The previous hash involves the calculation of a new hash. In this way, if anyone wants to temper a hash block, he has to modify all blocks after the tempered one. Otherwise, everybody who checks the blocks would discover that the data was modified since it cannot be assigned the right hash after calculating the tempered hash.

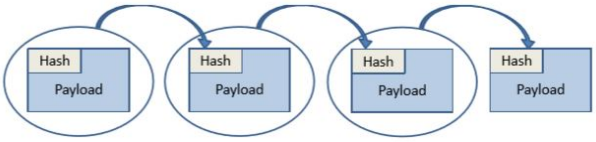

Fig. 2. Hashchain [11]

- Transaction: A transaction in blockchain technology is like a transaction between two accounts in banking, just with the difference that it can be seen by everybody on the net. A transaction consists of public addresses of all the input and output coins, and the signature of the input coins, which allow miners to verify the validity of the transactions without having to reveal their private keys. After miners verified a transaction, they will use the hash function to hash the transaction and combined them to form a data block. After they accessed a certain scale number in this block and solved the hash puzzle, they contain the block in the blockchain.
- Miners: The miners in a blockchain system have mainly two tasks: to verify the transactions and solve the puzzle problem.

2) What is blockchain Technology: Blockchain is mainly a distributed database technology.

In blockchain technology, a growing list of record is called a block. A block is linked together with other blocks in the chain by a hash function. Each data block consists of a hash and a payload. The hash of each block is calculated out of the whole previous block. The payload in each block is arbitrary data [11]. In order to mine a new block, the miner has to pool the data together and calculate a hash with the previous block, similarly to the hashchain. Nevertheless, there can be a hashchain inside a block of a blockchain, which implies, the data in a block are connected with hash. Besides that, all the blocks don't necessarily have to be stored together physically.

II. Authentication and Access Control

Authentication and access control are two commonly known methods to address the security and privacy problems in networks. They aim at preventing the unauthorized users to access information, and at preventing the legitimate users to access

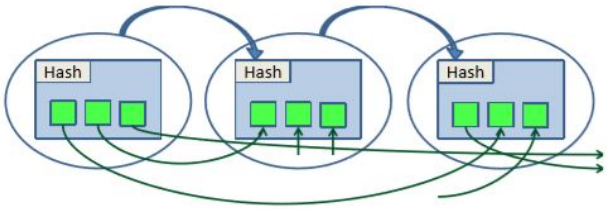

Fig. 3. Blockchain [11]

information in an unauthorized way. Likewise, they should assure that an unauthorized user can access the information in an advance authorized manner.

A. Authentication

As was mentioned in the introduction of three layers of IoT, the sensors produce the basic information for IoT network. But the users should decide who else can access the information. The first step is that a sensor should be able to recognize the illegitimate user, who is not allowed to give it commands. This action is called authentication. It is a communication protocol processing procedure [9]. When an information query arises, the sensor should authenticate the user(authenticated querying). It should satisfy the following properties [3]:

- Safety: If a sensor s processes the query q, then q was posted by a legitimate user U.
- Liveness: Any query q posted by a legitimate user U is processed by at least all sensors s$\in S_q$, where S_q is the set of sensors which must process the query in order to give the required answer to the user.

The establishments and distribution are the main tasks of entity authentication. There are currently two known algorithms to implement the authentication - Public Key Cryptography (PKC) based and Secret Key Cryptography (SKC) based.

Public Key Cryptography based implementation uses a pair of keys: a public key and private key. The public key verifies that a holder of the paired private key sent and encrypted the message. The private key allows the holder to decrypt the message that encrypted by the public key. However, because of its power-hungry, the sensors have to communicate with each other using symmetric cryptography. There are several PKC schemes and systems for authentication like Elliptic curve cryptosystems (ECC), Rabin's Scheme, NtruEncrypt and others.

Secret Key Cryptography based implementation uses a secret algorithm that employs two same key to encrypt and decrypt data. A SKC authentication system should provide a pair-wise key for all possible communications to all relevant parties. A sensor selects a shared key and physically transmits it to another communicator. The communicator then uses the shared key to encrypt and decrypt the data when it communicates with the sensor which sent it the key. If it wants to communicates with a third party, then they need another shared key between them. The problem with SKC is thus how to securely get shared keys and keep them secure after that. Mostly for these reasons, the PKC is now often used.

B. Access Control

Access control mechanism in IoT environment prevent illegal objects and users from accessing the resources.

- Centralized approach

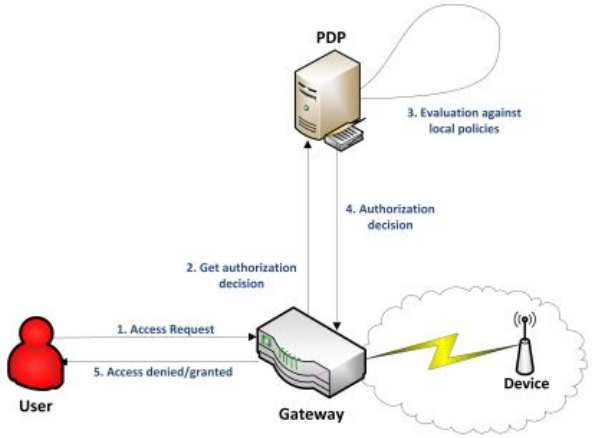

Fig. 4. Structure of Centralized approach [8]

In a centralized system, a central entity or central PDP (Policy Decision Point) is employed to externalize the access control logic, which is responsible for filtering access requests based on their authorization policies. As is shown in Figure 4. A PDP or such a central entity connects directly with a gateway, receives the authorization requests and then sends the authorization decision back after it has been determined. An authorization request arises when a user wants to access the data provided by the devices.

The advantage of the centralized approach is that the access control logic in an entity is not constrained by resources, compared to end-device with limited capabilities. However on the other hand, some problems exist with a centralized approach. Firstly, it lacks contextual information related to an end-device itself, while the contextual information is always highly relevant in many IoT scenarios. Secondly, the end-to-end security and the privacy of requester are compromised since a central entity makes the decision and reads the content of the access query. Thirdly, the trust of information providers and consumers with that entity needs to be managed. Finally, since the central entity stores all the information and has the power of their management, it's the most vulnerable point of the whole centralized system. If this center point broke down, the whole network would be paralyzed. Likewise, the central unit is an obvious target for potential attackers.

- Centralized and contextual approach

  In the centralized and contextual approach, end-devices participate in the access control decisions shown in Figure 5, in order to overcome the drawback that the centralized cannot provide a contextually relevant environment. For example in a situation of a medical emergency, the sensors inside the patient's body have to provide information

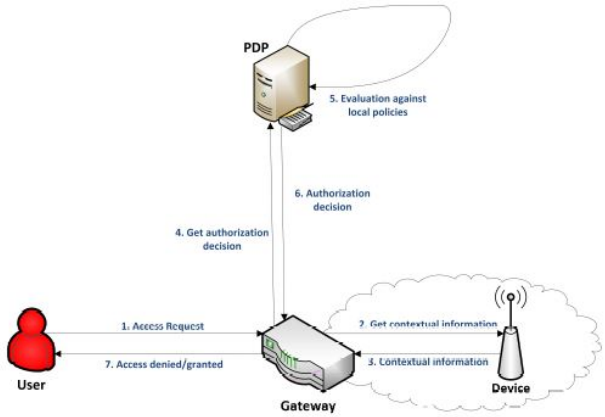

Fig. 5. Structure of Centralized and contextual approach [8]

for all the hospital staff instead of his or her attending doctor. It requires then some specific information, like the light level, heartbeat rate, CO_2 level etc. Under such situation, the end-devices are required to deliver the contextual information.

Hence, another access control mechanism performed by end-devices is needed. However, a trust relationship is assumed between the devices and the central entity. Following the previous approach, this trust establishment might be unfeasible in many of the scenarios foreseen for the IoT [8]. Besides, a delay problem arises when the information flows from end-devices to the central entity. Therefore, the value obtained by the end-device is different at the time of making the authorization decision. Finally, end-to-end security cannot be achieved [8].

- distributed approach

Fig. 6. Structure of distributed approach [8]

Compared to the two centralized approaches, the access control logic is embedded into the end-device in the distributed architectures. The end-devices are also called "smart things" or "smart objects" under such system since every device can acquire, process and send information on their own. This implies that, the end-devices can carry out the authorization process without the involvement of any central entity. As shown in Figure 6, the device determines the result itself and sends the authorization decision to the user directly.

The advantages of a distributed approach are obvious. First of all, end-devices are not passive any more, they are being smart to manage their information by themselves. Secondly, devices are able to send information when it is necessary since we do not want any central entity governs devices. In addition, the removal of intermediate entities enhances the security of end-to-end devices during the

access control. Finally, it simplifies the process of information exchange among end-devices.
The greatest disadvantage is coursed by an inability to provide the devices with access control logic. The native features of traditional models like RBAC and ABAC make their implementation unfeasible in resource-constrained devices. However, a distributed approach should be addressed more in-depth by analyzing the viability of different access control models or defining new proposals that meet the requirements of a distributed access control approach [8]. This is also the case from the security point of view. Protocols should guarantee basic security properties. However, the current solutions are based on cryptographic primitives, such as symmetric-key cryptography, connected to a high computational cost. The cost burden is incompatible with the principle of scalability of IoT scenarios. For this reason, a feasible and effective access control mechanism should be made by defining optimized public-key cryptographic.
Compare with the most access control solutions with centralized authorities, a decentralized access control solution can prevent governments, manufacturers, or service providers to gain unauthorized access to control devices by collecting and analyzing user's data. [12] There is quite often to hear that a service provider forwardly leaks their user's private information. It will never be safe to share our personal information with third parties. And a decentralized access control solution allows the user to control and master their own privacy and security requirement in mind. After over 20 years of scientific research, there have been significant advances in the fields of cryptography and decentralized computer networks, resulting in the emergence of a new technology —known as the blockchain—which has the potential to fundamentally shift our notions of centralized authority. [12]

III. Blockchain-based systems

A. Blockchain-based access control Framework for IoT

The presented advantages of decentralized authentication and the need for the security and privacy of IoT motivated, Aafaf Ouaddah, Anas Abou Elkalam and Abdellah Ait Ouahman to implement a distributed access control frame based on blockchain technology named FairAccess. [12] They also pointed out their new types of transactions, used for grant, get, delegate, and revoke access. The framework carries out the main characters to make user-driven. Only user has his own data, and only he can control the data and decide how it should flow. Using blockchain technology in this access control framework ensures that nobody could systematically be forced to loose control over his own data.

In FairAccess framework, a blockchain mechanism defines the low level functions to enforce policies and define how access requests are evaluated against those policies. [12] Below, a more detailed list is given.

- A bitcoin-like address, the public key, publishes a identity to every interacting entity.
- Blockchain is considered in the framework as a policy retrieval point. this means that every access control policy will be represented as a pair(resource, requester), and stored as transactions. It is used as a ledger keeping track and ensure the validity of access transaction among the interacting organization.
- Smart contract/scripting language expresses access control policies.
- Authorization token is defined as a digital signature that represents the access right or the entitlement. The authorization token is stored in a data field in a transaction, encrypted using the in-built cryptocurrency public/private key mechanisms.

Further, the framework building blocks are presented.

- A static description of framework building Blocks
  - User
    A user can be either a resource owner of a supplier organization or a requester from a requesting organization. The users communicate with each other through the transaction with their public keys.
  - Wallet
    A wallet is a data block which contains the Authorization policies, transactions and addresses, and is used for signing transactions, registering and identifying resources with their keys, and asking for access. In FairAccess framework, a wallet can specifically be an authorization manager point(AMP) which is a centralized entity for each organization and manages authentication and authorization data for a resource. It could be a web or mobile application. The resource owner can register the resources to be protected and define access control policies through it.
    A wallet has mainly three functions: generates keys and addresses, transforms access control policies to a transaction and broadcasts them later to the network and validates transactions.
  - Address
    An address is used to address the identities of entities in the network, which are public and shared in the network. In FairAccess, an address is a hash of an Elliptic Curve Digital Signature Algorithm public key and a corresponding private key. It is the basis of decentralized trust and control and the cryptographic-proof security.
  - Transactions
    A transaction is a fundamental building block which are actually unspent transaction outputs (UTXO) in the bitcoin system. It records the inputs and output addresses (Vin[] and Vout[]) and the amounts of a transaction. In FairAccess, we do not transfer bitcoin but access token where a token is an access right defined by the creator of the transaction to the receiver of the transaction to access Resources identified by their address in the transaction [12]. The UTXO corresponds to an access taken, that is recognized by the entire network as an access right defined by

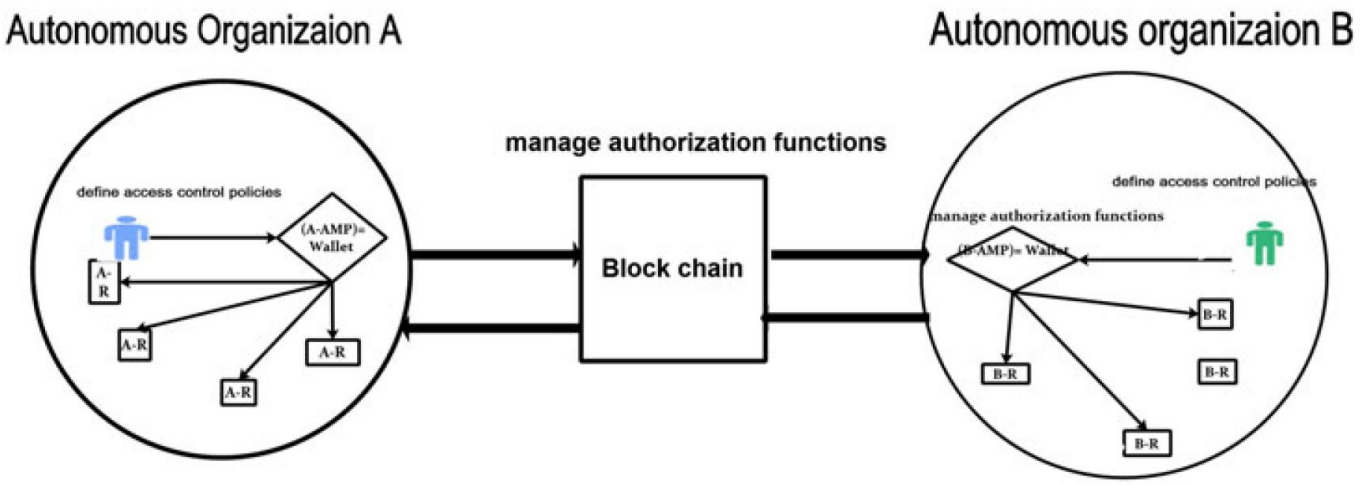

Fig. 7. FairAccess architecture overview [12]

the creator of the transaction to the receiver of the transaction to access resources identified by their addresses in the transaction. [12] In the paper, there are two kinds of access tokens consumed and created in a chain of transactions. Namely, GrantAccess, a Tokenbase transaction where a new token is generated when a resource owner defines an access policy, and another kind of a regular transaction, that can either be a GetAccess or DelegateAccess transaction. A GetAccess transaction will be generated, when a requester spends a token that he obtains from a previous GrantAccess transaction, in order to access a resource identified with an address if it fulfils the access control conditions. And a DelegateAccess occurs when a requester delegates access by transferring token that he owns to another owner.

Besides, every access token is encrypted with public key of the requesting party to which the token is designated, and only requesting entity to which the token is created can decrypt it with its private key.

An input vector Vin[] has the following parts: *Index* which notes the index of the input in this vector. *Reference to previous output*(a hash, index tuple) which refers to the IDx and a sequence number of the recorded token in the blockchain. However this does not mean anything in a GrantAccess transaction because it generates a new access token. *Resource address*or the address of the requested resource. *Unlocking scripts*, a script satisfying the access control policies.

An output vector Vout[] consists of the following elements: *Index*, which stands for the index of the output in the vector. *Value* means the transaction fee. *TKN*, represents an unspent access token. *Requester address*, is a party requesting to access the resource.*Locking script*, the locked token, that dedicates the access control policies they should meet to spend the token.

The blockchain in FairAccess is used as a database that stores all transactions and access policies that were generated by users or nodes (a full node in the network is a collection of functions: routing, the blockchain database, mining, and wallet services [12], and is used to broadcast the transactions. It provides a working proof of a decentralized trust, since all transactions group in a public ledger that everyone can access. This ensures a complete consensus.

- Authorization model, architecture and mechanism In this section, a dynamic description of FairAccess authorization framework is presented, In particular: how the framework works with six authorization functionalities by dealing with address, transaction and access control policies. Firstly, I will slightly describe what the six functionalities are, and then demonstrate how the framework gathers them together to achieve secure authorization.
  - Transaction validation protocol: is executed by nodes in the network when a transaction is received. All nodes validate independently every transaction before propagating it further, so that an attacked transaction does not pass. A signed transaction consists of two parts, the transaction itself and a hash of this transaction data, which is encrypted with the private key of the sender.
  - Registering a new resource: the main technology in blockchain is the digital signature, which means a pair of keys. The resources in the system are associated with a corresponding address. We use a one-way cryptographic function - elliptic curve multiplication to generate a public key, and from the public key, we use a one-way cryptographic hash function to generate a private key. A private key is then used to (i) create a signature to prove ownership of resources, (ii) to control access (by defining access control policies) in GrantAccess, and (iii) to prove the possession of access token in delegate access.
  - Grant access to a requester through a GrantAccess transaction type: GrantAccess is the most frequently used transaction in this framework. It defines the

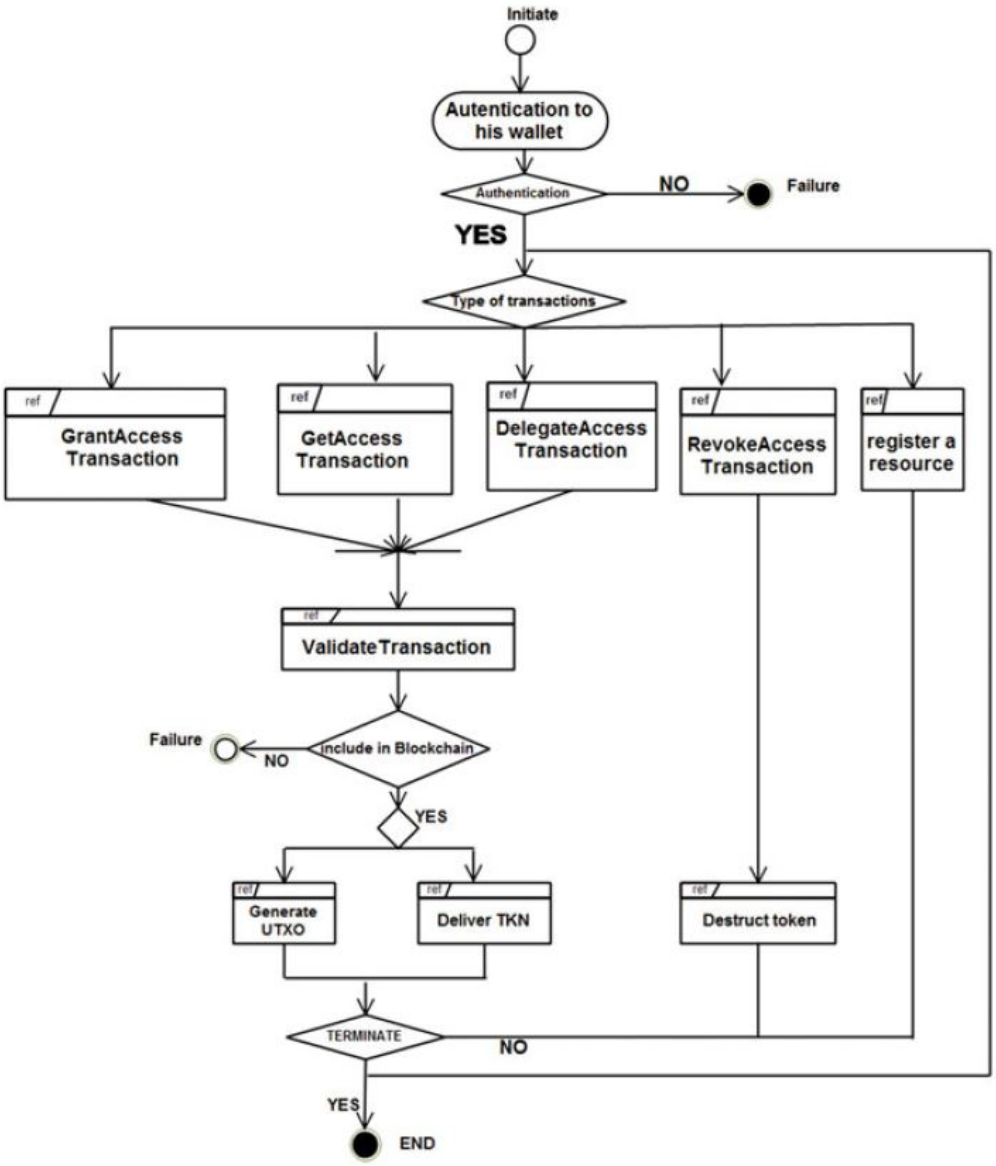

Fig. 8. FairAccess interactive overview diagram [12]

access policies in a form of GrantAccess transaction protocol on a level of the couple(ResourceAddress rs, RequestAddress rq). The wallet then transforms this access control policy to a scripting language and generates a GrantAccess transaction which should be signed by resource owner. It further propagates to the network. After that, nodes verify the transactions according to transaction validation protocol. Finally, transactions are valid and recorded in the blockchain. If the transaction is not valid, a notification is sent to the resource owner. The policy actually tells everyone, that a token is delivered from A.pk to B.pk for a requested resource. Nevertheless, the requester has to unlock the access condition and decrypt the token with his private key by proving the fulfilment of access condition in a new transaction called GetAccess.

– Access to the resource through a GetAccess transaction type: This transaction entails unlocking the script and spending the value of the TKN in a GrantAccess transaction. The requester uses GetAccess transaction to prove to the network that he fulfils access condition. The requester can spend the token either by accessing the resource or delegating access transaction. In the first case, the requester scans his token database created by wallet by scanning the blockchain and collecting all tokens associated to the client address. If there is a token corresponding to the resource, the system generates a GetAccess transaction. Otherwise it sends a request to the resource owner containing his address. When the requester meets the access condition, an unlocking script is generated and the whole transaction broadcasts to the network. The nodes then verify and validate the

transaction and decide to accept it to blockchain or to reject it. Once the transaction appears in the blockchain, it is implied the client has fulfilled the access condition. The token can be then delivered to him. In the second case, the delegator generates a DelegateAccess with the public key of the real requester instead of him in the output. After it is validated, the transaction then appears in the blockchain and the wallet of the real requester. The real requester can create a GetAccess like a delegator to access the resource if he fulfils the access condition.

- Delegate access through a DelegateAccess transaction: fully corresponds to the second case described above.
- Revoke access through a Revoke Access transaction type: The resource owner can revoke access by introducing the time-validity of the authorization token as a field to be checked in the token. [12]

The resource uses public key as an address for transactions, and the interacting sensors can sense a public key to obtain an address to receive access request without knowing the private key of the party which avoids the vulnerability. However, whether the request would be accepted or denied still depends on the resource owner, who can sign the transaction with the private key. And the blockchain in FairAccess is a centralized database with different types of transactions, that cooperate together to achieve the security and privacy of the framework indicated above.

B. The Case Study of a Smart Home

Before knowing how a smart home works, I would like to highlight the different meanings that three core Components mean from blockchain to smart home.

1) Transactions:
Transactions in a smart home system stands for the communications between local smart devices or overlay nodes.
2) Local blockchain
Unlike usual blockchain in the blockchain technology, the blockchain in a smart home system will be only stored as local private blockchain and managed by a local miner. There are two headers in a local blockchain, namely, block structure header and policy header. The block header contains the hash of the previous block, while policy header has an authorizing list. The policy header is shown in Figure 9: Which define the permission of an action that a requester requests for a device. The block header(shown in Figure 10) defines the structure of transactions in the block. Which shows the order, type and the content of a transaction arising from a device.
3) Home Miner
Compare with the miners, the nodes from a blockchain technology system, focusing in solving puzzle problem and verifying a transaction, a home miner is a device that centrally processes incoming and outgoing transactions to and from the smart home. [7]

#	Requester	Requeste for	Device ID	Action
1	kdhqipab21	Access	4	Allow
2	All	All	5	Deny
3	aoqho42d1	Store	2	Allow
4	3	Store Cloud	3	Allow

Fig. 9. Policy header in a local blockchain [7]

Previous Transaction	Transaction Number	Device ID	Transaction Type	Corresponding multisig Transaction
N = Genesis T.			Genesis : 0 Access: 1 Store : 2 Monitor : 3	If any (for keeping signature of requester).

Fig. 10. Block header in a local blockchain [7]

The next paragraphs further describe what a smart home is, how it works with a blockchain mechanisms and why it is worth to employ a blockchain mechanisms in smart home system. The case study is based on a research paper - Blockchain in Internet of Things: Challenges and Solutions from Ali Dorri, Salil S, Kanhere and Raja Jurdak.

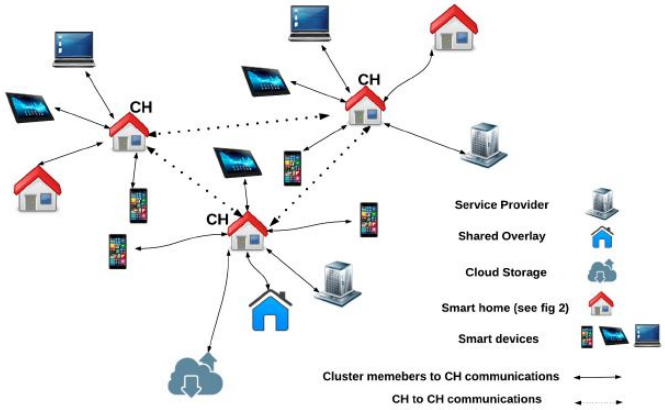

Fig. 11. A smart home [7]

1) Smart Home:
A smart home is also comprised of three parts: smart devices, local blockchain and local storage. The smart devices are the end-devices located in the home. Local blockchain is mined and stored by one or more resource-capable device, such as a hub or home computer. However, a local blockchain is central, managed by its owner. All transaction pertaining to a particular device are chained together. The owner manages a new device by adding a transaction in the chain like includes a Bitcoin in a chain. And the owner can remove a record from the ledger to delete an existing device. The local blockchain

has a policy header, which is an access control list that allows the owner to control all transactions happening in her home [7] and the most updated one will be sent to the front. The permission of data exchange among devices can only arise when owner allows them to do that with a shared key based on the generalized Diffie-Hellman algorithm. But unlike Bitcoin blockchain, each block is mined and appended to blockchain without POW, the miner adds a pointer to the previous block and copies the policy in the previous block header to new block and chains the block to the blockchain, and a transaction will not be verified since we want a lightweight blockchain. The local storage in a home used to store data locally such as a local backup drive.

2) Overlay Network:
The overlay network is like a peer-to-peer network in Bitcoin. The nodes are smart home miners, other high devices in the home or user's smartphone or personal computer. Each node uses Tor to connect to overlay network for additional anonymity at IP-layer. a Tor is a free software and an open network that helps user defend against traffic analysis, a form of network surveillance that threatens personal freedom and privacy, confidential business activities and relationships, and state security [1]. The nodes in the overlay network are grouped in clusters with a cluster head in order to decrease network overhead and delay. The clusters can group in other clusters if they experience excessive delays. In addition, they can also elect a new cluster head at any time. A cluster head is responsible for maintaining three lists:

- Public key of requests: All public keys on the list can access data for smart homes connected to this cluster.
- Public key of requestees: All public keys on the list are allowed to be accessed.
- Forward list: a list of transactions sent for other cluster heads in the network.

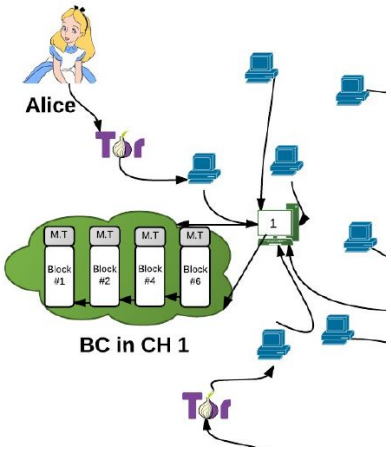

Fig. 12. Local blockchain in network [7]

An overlay network blockchain is kept by all cluster heads in the overlay network. Since a user can have many homes and wants to manage them together, a shared overlay consisting of many devices from multiple homes can be formed.

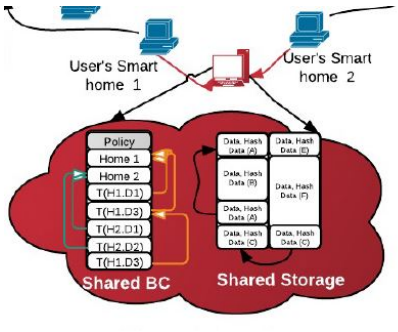

Fig. 13. overview of a share overlay [7]

3) Cloud Storage
In a smart home, some applications may require to store their data in cloud storage. A third party Service Provider(SP) can therefore be allowed to access acquired stored data and provide other services. The cloud storage groups user's data in identical blocks with a unique block-number for authentication. If the storage can locate data with a block-number and a hash, the user is authenticated. If new transactions are added into a block, the new block-number is encrypted using a shared key which means only the user knows the block-number. Users can create either different ledgers or a single ledger for all of their devices.

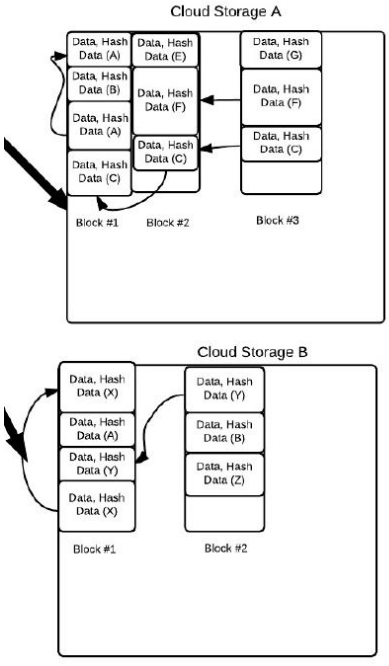

Fig. 14. Structure of cloud storage [6]

4) Transaction Handling
The communications between local devices and nodes are transactions. There are mainly three options in

transaction handling,

- Storing:
  Data in smart home can be stored in local, shared or cloud storage. After having checked the validity of the transaction and space of the storage and comparison of hashes, data packets are stored and new block-number is encrypted with a share key, and sent to miner after that. Moreover, the signed hash of data is signed by the storage and sent to overlay network to be mined in the overlay blockchain.
- Accessing:
  If a service provider requests to access the stored data, it has to create and sign a multisig transaction, signed by the requester(service provider) and the requestee(miner of smart home) and send it to its own cluster head. The cluster head then checks both lists of public keys. If either the multisig transaction's requester is in cluster head requester's PK list or it's requestee is on its requestee's PK list, then it broadcasts the transaction to its own cluster. Otherwise, the transaction is broadcasted to other cluster heads and the PK of requester is put in the forward list. The miner has to check the policy in the local blockchain to verify a service provider when he received a multisig transaction. If the permission has been granted previously by the user, the miner requests packets from the storage, encrypts them with requester's public key and sends them to the requester. After having accessing the data from storage, the miner should store multisig transaction in local blockchain, and can send it to a random set of cluster heads to be stored in the overlay network. In this case, it should be proved that the data was sent by the user. However, it is not necessary at all to place the multisig transaction in the overlay network; for example to prevent attackers from stealing a real world identity.,

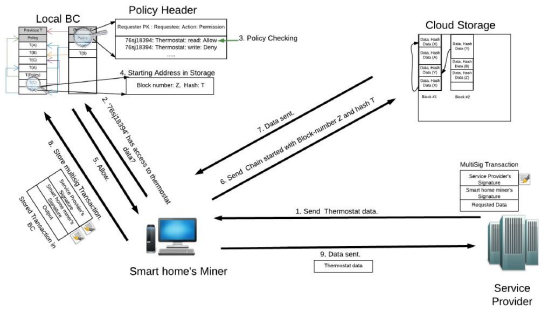

Fig. 15. Accessing process(with a smart device "Thermostat") [6]

- Monitoring:
  A monitor transaction is an option of the user who wants to access certain information of a device real-time. The process is essentially similar to an access transaction which ensures the security of data by looking into policy headers in a local blockchain.

In this paper, the authors simulated the blockchain based smart home system in reality. They evaluated the smart home under a blockchain mechanism and a baseline method, and compared their performance in terms of the packet overhead, time overhead and energy consumption, which refers to the length of transmitted packets, the processing time for each transaction and the consumed energy by the miner for handling transactions, respectively.

From the result of the packet flow, we can see that a baseline system transmits the packets with a length of five bytes in any transmission. In contrast, the blockchain based one increased the payload to 16 bytes packet from devices or the cloud to the miner and 36 bytes from the miner to the cloud to accomplish the same action. However, considering the lower layer headers (i.e. 6Low-PAN), the increase in the data payload has a relatively small effect. [7]

From the time overhead perspective, they found that a blockchain based method takes 140% to 150% time consumption of the baseline method in periodic store transaction, query based store transaction and access transaction. However, the time that a blockchain based method takes for a query based store transaction is around 68 ms, which is acceptable.

The result of energy consumption demonstrates that blockchain based method uses 0.007 to 0.008 mj of energy more than a baseline method when executing the three above mentioned actions. To sum up, blockchain base smart home system uses slightly more energy and time to accomplish tasks compared to a baseline method. This is a consequence of a long transmitted data packets due to security and privacy it offers.

IV. Conclusions

The threat of an attack to a system will never vanish, because there simply no foolproof system exist. We have discussed the security challenges from three layers' aspects of IoT, introduced a variety of attacks and possible countermeasures in every layer. Current researches are mainly focusing on setting a more robust authentication and access control. By comparing centralized, centralized-contextual and decentralized approaches for access control, we concluded that the decentralized approach may be the best suited approach for current state and application of IoT. This is the case especially because of its distributedness feature. The framework FairAccess, embedded within a blockchain technology, implemented quite a secure and transparent access control tool. However, it seems like a heavy-handed framework which can cause problems in a real time environment. The framework embedded within a lightweight blockchain technology, used in the smart home case study, seems a smart way to achieve security and privacy, although it raises the consumption of energy than a traditional method. Despite the slight inefficiency, it still seems to be a very fair trade-off..

References

[1] T. Project.
[2] M. Abomhara and G. M. Køien. Security and privacy in the Internet of Things: Current status and open issues. 2014.

[3] Z. Benenson, N. Gedicke, and O. Ravio. Realizing Robust User Authentication in Sensor Networks.

[4] M. A. Bhabad and S. T. Bagade. Internet of Things: Architecture, Security Issues and Countermeasureses. 2015.

[5] M. Ciampa. *CompTIA Security+ Guide to Network Security Fundamentals, 6th Edition*. 2018.

[6] A. Dorri, S. S. Kanhere, R. Judak, and P. Gauravaram. Blockchain for IoT Security and Privacy: The Case Study of a Smart Home. *Conference Paper*, 2017.

[7] A. Dorri, S. S. Kanhere, and R. Jurdak. Blockchain in Internet of Things: Challenges and Solutions.

[8] J. L. Hernéndez-Ramos, A. J. Jara, L. Marín, and A. F. Skarmeta. Distributed Capability-based Access Control for the Internet of Things.

[9] J. Liu, Y. Xiao, and C. L. P. Chen. Authentication and Access Control in the Internet of Things.

[10] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zulkernan. Internet of Thing (IoT): Security: Current Status, Challenages and Prospective Measures.

[11] O. Mazonka. Blockchain: Simple Explanation. 2016.

[12] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman. FairAccess: a new Blockchain-based access control framework for the Internet of Things.

[13] R. P. Padhy, M. R. Patra, and S. C. Satapathy. Cloud Computing: Security Issues and Research Challenges.

[14] D. G. Padmavathi and M. D. Shanmugapriya. A survey of ATtacks, Security Mechanisms and Challenges in Wireless Sensor Networks. 2009.

[15] X. Xu. Study on Security Problems and Key Technologies of The Internet of Things. 2013.