Towards Lightweight Authorisation of IoT-Oriented Smart-Farms using a Self-Healing Consensus Mechanism

Steph Rudd, Hamish Cunningham
University of Sheffield
Sheffield, United Kingdom
lia08sr, h.cunningham@sheffield.ac.uk

***Abstract*—This paper proposes a novel consensus mechanism towards access control in networks populated by constrained devices. The research explores existing mechanisms in the distributed technology blockchain, towards limiting scalability of data cumulation for microcontrollers, specifically the ESP32. The work utilises security functions native to the featured ESP32, for a more granular approach than binary authorisation; security solutions such as the de facto standard TLS adopts an 'in or out' approach, and this can invite numerous problems. The design proposes a hybrid design to produce a robust solution that suffers from neither the vulnerabilities of centralisation or data aggregation of distribution. The decentralised proposal is then tested using formal verification; model-checking using the AVISPA and SPAN tools. The research concludes a safe design.**

I. Introduction

Envisioned as a clean alternative to global food and environmental issues, smart agriculture has been highlighted as playing an important role in efficient food production [1]. Within the realm of smart agriculture, aquaponics presents a symbiotic relationship between plants and fish requiring less fertiliser, by utilising fish effluent instead [2].

However, traditional network topologies pose the vulnerability of centralisation; a single point of failure defined by a central server [3]. In contrast, distributed data storage presents a scalability challenge with the limited storage of constrained devices, particularly in lightweight, battery-powered situations [4]. Ideally, processing and storage should be kept to a minimum, to prevent further burden of energy consumption and subsequent pollution that the modern data deluge presents [5].

Security is a fundamental requirement to assert business continuity [6], and generally the interpretation of security refers to authentication between devices, and the subsequent secure channel for messaging following authentication. However, default authorisation (or access control), generally allows any device onto a network given the correct password. This approach is relatively black-listing; participants are generally welcome if they have basic entry requirements.

Basic entry requirements can be provided by any node or living subject with such knowledge, and this opens up the threat landscape enormously, such as to malicious bots [7]. Malicious bots permit the large-scale interference of decision making by targeted algorithms or network seizure. Perhaps a tiered method of device authorisation would be where participants are generally unwelcome onto the network if they have basic entry requirements, are required to undertake further interrogation, and is therefore more comparable to white-listing than black-listing.

Distributed Ledger Technologies (DLT), [8], such as blockchain, present valuable methods of decentralised data exchanges between multiple nodes [9], based on consensus between groups of nodes rather than individual subjects or devices. When consensus is reached, data will be accepted as part of a block and placed on the chain, accumulating as recorded transactions. Although cumulating data is not ideal for resource-constrained applications, the consensus mechanism is helpful for granular and autonomous authorisation; nodes run by policy can therefore govern and self-heal a network beginning with, but not limited to, network passwords.

This research proposes a hierarchy-based authorisation model in which a consensus mechanism allows limited devices based on rank and network privilege, and unlimited devices without such. Data chains are ephemeral and limited, based on clustering of those within range [10], towards a decentralised model. This hybrid between centralised and decentralised security also aims to be agnostic, protecting heterogeneous messages and devices described as challenging [11].

The contributions of this research are:

1) Low processing and storage: no databases, lists, central servers or network-wide data chain.
2) Self-healing: missing hierarchy positions are recognised and replaced between clusters.
3) Limited chain: limited hierarchy positions between clusters restrict chain contents.
4) Agnostic application: heterogeneous devices benefit from using existing security functions.
5) Consensus: approval requires network password and 'good reason' for joining.

This paper proceeds with a problem background, related work in the field, followed by design and testing, model-checking results, and concluding with the safety of the design.

II. Problem Background

The heterogeneous nature of IoT networks presents three concepts difficult to cater for; constrained power and process-

ISSN 2305-7254

ing in devices, decentralised topology, and the challenges of heterogeneity.

A. IoT Device Constraint

When IoT devices are to be networked, the challenge is always with power, processing, and storage [12]. Traditionally when a network is built, there is a central server surrounded by nodes which rely on that server for decisions pertaining to data access - particularly network roles, who can access and change the states of what data, when, and why. Such a system is referred to as Access Control (AC), a series of roles and rules, depicting read, write and execute privileges amongst data objects [13]. This central server is generally responsible for storing a lot of functional data that the nodes relay to it, and an IoT network is only different in the sense that it is designed to be autonomous, whereas typical computer networks have users at the end of them. There are further ideologies that can alter the way of things with IoT [14], setting it apart from a traditional network. Firstly, they can be independent of the internet because of low-powered protocols such as BLE, LoRaWAN, and MQTT [15]. This provides for off-grid placements of devices, beyond the range of regular WiFi reception - BLE has a range of around 100 metres, and much more than that when meshed. Meshing and autonomy promotes scalability, but discourages the use of a single server due to range, data growth, and a central point of vulnerability.

Some topologies operate in clusters of devices with a central node of their own, working in small numbers of in-range devices to take advantage of the low-energy protocols and data distribution. There is also the more recent development of Distributed Ledger Technologies (DLT), such as blockchains [16], hashgraphs [17], and Directed Acyclic Graphs (DAG) [18], which verify and store transaction data based on consensus mechanisms and smart-contracts. DLTs are the most robust, scalable and reliable of data-centric topologies, but the big data scalability would be the failure of IoT devices alone, and defeat the notion of carbon-negativity if additional storage facilities were provided.

B. Typical applications

Most mature AC systems are based on roles or rules, and utilise Access Control Lists (ACL), and are not dissimilar from the traditional Relational Database Management System (RDBMS) [19], mapping usernames to privilege levels throughout known sets of data. ACLs present a few problems to the IoT network. Firstly they are central, and have to be kept somewhere that can be accessed by every node or microcontroller, then there's the requirement to know all the data available, and somehow map it to people or devices that will read, write, or execute it - even during the 1990's before the big data we process now, this was becoming an issue. Relational data management is difficult with IoT since it restricts autonomy, and becomes increasingly complex as new developments are integrated. Finally there's the duplication of data - for example if this list covers all possible relationships (for which some M:N relationships will require additional tables for normalisation) [20], this is a big expectation for limited storage space. We consider other, mature AC systems:

DACS: Distributed Access Control System, is rule-based, and operates on the identity of users credentials within the organisation. This type of account setup would apply to every microcontroller in a network, taking time and resources to create large uniqueness for what should be an automated, preferably self-healing, relatively generic node.

DAC: Discretionary Access Control, is also rule-based, and assigns access rights based on rules specified by users, utilising ACLs and capability tables - rows with subjects, and columns with objects.

MAC: Mandatory Access Control, is considered the most strict of authorisation systems, and uses hierarchy to apply granular levels of access within a varying number of tiers, or 'security labels' often under the classifications of low, medium or high. The MAC system also permits access on a 'need to know basis', reminiscent of the GDPR's 'necessary processing', and thus pertinent to the topic of privacy. MAC maps user accounts and credentials to security levels and departments using ACLs, and requires constant updating of data objects and labels due to the high level of security it provides.

RBAC: Role-Based Access Control, operates on roles, rather than individual user accounts, using the 'least privilege' possible for the operative to fulfill the job, similar to MAC but not as strict.

PBAC / ABAC: Policy, or Attribute-Based Access Control, is whereby access rights are granted using policies to combine resource, user, object or environment attributes. This is a Boolean-centric model using 'if' and 'then' statements about who the request is made by, the action they request and for what resource. This model does not use pre-defined roles or lists.

Other applications include capability certificates provided by policy engines, Elliptic Curve Cryptography (ECC), and based authorisation certificates [21], but utilising certificates requires some form of authority by which to negotiate terms such as signatures, origin, roots and leaves - a problem similar to ACLs and centralisation. Similarly, keys have been used to update access and permit certain users - but a centralised system must also be employed to generate and monitor fair usage [22]. [23] presented a hierarchical authorisation system in which users of similar privilege levels are placed in the same group. This type of trust-organised placement perhaps makes more sense in IoT, where greater trust correlates to older, probably original devices.

C. Heterogeneity

The essence of IoT heterogeneity pertains to the differences between communications and capabilities of components belonging to the same network - a big challenge in the IoT domain is resolving relationships between devices, and adjusting our expectations [24]. Our expectations as end users towards handheld and desktop devices along with internet connectivity have increased as the capacity of such have developed in

parallel, IoT development contradicts that model. IoT presents constrained power, processing and storage, and so opposes 'big data'. In many ways this is positive - moving away from a data deluge, away from servers, hardware storage and energy consumption are very environmentally beneficial, and may work towards changing our expectations of technology improvements towards a 'greener IoT [25].

With communications protocols such as Bluetooth Low Energy (BLE), Long-Range Wireless Area Network (LoRaWAN), Message Queuing Telemetry Transport (MQTT), Zigbee and numerous other examples, IoT protocols are designed to match the capabilities of the boards to which they are native. This is very helpful, since microcontrollers supporting BLE, LoRaWAN etc, are designed with those small packet sizes in mind - sensor readings and actuation code to change the environment around us in any application - domestic, industrial, agricultural and so forth. The caveat comes in when implementing security [26], [27]. Transport Layer Security (TLS), has been the de facto standard for decades. TLS is strong, and promotes public-key cryptography over applications in banking, e-commerce, ATMs, chip-and-pin, and fortunately, the majority of communications protocols for constrained devices; notably BLE, MQTT, and LoRaWAN. At first glance, the principles underpinning TLS practice would be most suitable for an AC system, because those same functions, in some way, will support the message and authentication of the same aquaponics system, thus minimising the keys, identifiers and cryptography processing of a larger application. However, adjustments must be made for a successful transformation over the state of TLS in its most contemporary form. The asymmetric key length requirements and symmetric ciphers stipulated by TLS 1.3 are not IoT-friendly - neither are centralised role or rule lists, DLTs, or most other approaches to mature ACs.

III. Related work

This section considers literature that will influence the design. Considerations include privacy, network topologies, and consensus mechanisms.

A. Privacy and security

Access control and authentication are two main challenges in security and privacy requiring an immediate solution in the heterogeneous environment of IoT [28], proposed a capability-based authorisation model in which a token provides trust and a session key is produced to authenticate using Elliptic Curve Cryptography (ECC). Tokenisation has been introduced as ephemeral data for cryptocurrency transactions alongside various key exchange systems such as Diffie Hellman, RSA and the lighter ECC. Whereas tokens are not strictly cryptographic functions and are typically referred to as advanced pseudonymisation as a privacy function [29], they do limit the spread of data beyond a "need to know" basis, a fundamental concept of the GDPR [30].

Wherever possible in a constrained network, minimising the processing requirements of securitisation is beneficial - for power, time and subsequent carbon consumptions. As a result, privacy functions are preferable if they so address the minimal necessary processing data for a given task - and this leads naturally to hash functions [31], tokens [32], ephemeral chains [33], and minimising the use of multiple identifiers, such as using keys as identifiers. However, a network must still be secure, and this is where the use of keys is required at some point - either asymmetric or symmetric key exchange is used to encrypt data to a standard of security that cannot be guaranteed by privacy functions [34].

B. Topologies

Centralisation pertains to governance and processing using a central authority, usually for encryption key escrow, device identifiers, intrusion detection, and review and approval of new network devices. Organisations have typically used centralised services to provide X.509 certificates for banking and shopping gateways, and this has been a safe approach for a number of years - appropriate for servers that operate from a single or small number of locations. Centralised infrastructures are however, energy inefficient, usually rely on internet connectivity, must be active and online in order to be used, and are rendered vulnerable by a single point of failure [35].

DLTs, have demonstrated robust and accessible networking for cryptocurrencies [36], data storage [37], asset tokenisation [38], and e-commerce [39]. Although DLTs are generally constructed in graph and blockchain form, they all accumulate data to be operational - and this inevitably requires more storage and energy to operate.

Decentralised architectures introduce a hybrid design, in which there is no single point of failure, but a defined group of devices either shares the same data, or relies on its own central device. A decentralised infrastructure can be spread over a long range using any number of technologies with communicating central nodes being elected by voting systems using reputation, such as with P2P ranking such as for file sharing online [40].

A decentralised topology with limited device numbers in groups of nodes, and replicated, ephemeral chains operating on a lightweight consensus mechanism would provide a suitable hybrid topology for a constrained IoT network such as the aquaponics smart-farm.

C. Consensus mechanisms

Consensus mechanisms are fault-tolerant process used to achieve necessary agreement from a group for a single data transaction. The consent of multiple nodes then gives the permission for that transaction to reside as part of a data block on the chain. Below are examples of popular mechanisms:

Proof of Work (PoW), requires members of the network to solve mathematical puzzles to prevent anybody from occupying the system, is known as mining, and widely used in cryptocurrencies to prevent attacks such as 51% - but PoW is energy-intensive. Proof of Stake (PoS), select transaction validators based on proportion of ownership on the chain compared other members, and although does not use extreme

amounts of energy like PoW, it is criticised for 'blockchain oligarchy', and as such is vulnerable to the 51% attack. Proof of Elapsed Time (PoET), is where users wait a random period of time and the first to finish waiting attains leadership of the new block. PoET uses trusted code with trusted, known users on a permissioned, or non-anonymous, network. Practical Byzantine Fault Tolerance (PBFT), is based on the Byzantine Fault Tolerance algorithm in the late 1990's; a scenario in which General's gather around an enemy city and must communicate without interception. They gather around the city and send one message a time through a messenger, relying on cooperation and coordination, ignoring unauthorised influence. Proof of Importance (PoI), introduced a rating system to combat the 'oligarchy' issue of PoS in which the rich become richer. The rating system requires a minimum vested stake, a minimum transferred rate, a recent transaction bound by time limit, and valid transaction partners identified as separate users. The algorithm does not require complex computation, but rather a series of experience proofs to read through quickly. Proof of Capacity (PoC), is where the mining node's hard drive space is used to determine the mining rights on the network. Considered an improvement on PoW, it takes 40% of the time to produce a block by dedicating storage and processing before mining begins, and removes the mining conflict of the same puzzles by allowing different 'plot' routes for working through puzzles - so there is both hard drive plotting, and block mining. Proof of Authority (PoA), uses identity as a stake for reputation-based consensus, which means validators are not staking currency, but their reputation instead. PoA blockchains are therefore secured by random validating nodes as trustworthy entities. In supply chains and hierarchies, PoA is considered an effective and reasonable solution. Proof of Location (PoL), enables a devices physical location coordinates to be broadcast to the blockchain without relying on that particular device. Radio, GPS and BLE-enabled devices can assess the physical location of nearby devices, but must be encrypted for reliability.

Influences of 'necessary processing', privacy, decentralisation and minimal energy consumptions through lightweight consensus mechanisms influenced a design that could be later model-checked for secrecy.

IV. Design and model checking

With considerations of the problem and related work, the design proposes decentralisation, low consumption security functions, privacy principles, autonomous policy-oriented authorisation, and group consensus. This section presents the design:

A. Decentralised

Decentralisation in preference to centralisation or full distribution. Ideally, the topology should avoid duplication of the same data between all nodes, but allow duplication between associated nodes. Decentralisation introduces robustness unavailable in centralisation, as there is no central point of vulnerability, without the scalability issues of full distribution.

A decentralised topology of clustered devices is proposed, where clusters are arranged by the BLE range of up to 50m (100m is the maximum range), and each cluster contains a series of devices divided by rank and privilege:

TABLE I. Device Hierarchy

Rank	**Privileges**	**Max. Devices in Rank**	**Security**
1 Field Marshal	Read Write	2	Rank-shared symmetric key, does not advertise MAC or public key
2 General	Read Write Execute	3	Rank-shared symmetric key, does not advertise MAC or public key
3 Unranked	Read-only	Unlimited	No rank secret, advertises MAC and public key

This is a simple but granular design, beneficial for cloud and data governance [41], and requires three devices to operate properly; an active General, a passive FM for backup, and an Unranked to protect the General from any threat. This is positive for two reasons. In a range of systems, the higher privileged devices are few enough to be realistic, but robust enough to form a group consensus beyond the powers of unranked, new, unproven devices. Generals may execute changes in privilege levels to unranked devices, promoting them to General when necessary, but are limited to three. The idea of a hierarchy is not to allow widespread privilege, but minimise the privileges wherever possible so that the number of authorised devices remain intact, and the purpose of authorised devices in business continuity. Should the system suffer a majority loss, assuming that at least one General and a Field Marshal survives within BLE range, the whole authorisation system can self-heal. Now, privilege levels are purposefully kept simple:

1) Field Marshals: the highest and most protected part of the rank. They can only be accessed by a General or other FM, and do not have the power to grant authority. This is to protect them from manipulation, since they are the highest in the chain. There are two Field Marshals in every cluster just for insurance, although any system can be one. The term Field Marshal was used to depict a higher rank than the active General, but passive.
2) Generals: the most authoritarian because they are neither the top of the hierarchy, nor vulnerable to every potentially harmful device. The ability to execute means they can grant rank identity to new Field Marshals, new Generals, or unranked, unprivileged entities. There are more of them than Field Marshals since their workload is expected to be higher. Generals hold a secret shadow used to gain consensus from other cluster generals by joining all the shadows together. The term General is commonly used in cryptography literature such as in the aforementioned 'PBFT', and is an active role.

3) Unranked: the read-only privilege of unranked devices means that they can be exposed to a threat landscape without risking the integrity of the network structure. They can hold position as any system, freely act as a server for new client relationships, and seek authority from a General should they connect with a suitable new entity.

1) Low consumptions: Since there are many ways of implementing security between nodes, such as keys and certificates, this aim requires a consideration into the lightest application of security functions available on the ESP32. The objective here is for the least infrastructure possible and fewest messages between nodes for safe network access. As the system continues in its daily life cycle, it will utilise the four functions hosted on the ESP32 for authentication and message security. Although the authorisation system will also use a selection of these functions, it will be a separate process than those required on a daily basis. Authorisation is expected to be infrequent compared with the other processes, and ideally without conflict of other security keys and sensor-actuator tasks. The following functions are native to the dedicated hardware of the ESP32 and thus use as little energy as possible during authorisation:

MAC: Media Access Card is the 12-digit BLE address given to identify each ESP32, or each aquaponics system, in the aquaponics farm. The MAC address can be changed to a meaningful label describing geographical location or crop type.

Public key: used as asymmetric cryptography is order to encrypt a message dedicated only for that recipient, so that no other devices can decrypt it. Designed to be publicly available and traditionally kept in a database, the public key is requested on joining.

Private key: used as part of asymmetric cryptography for proving origin of the message by providing a digital signature. This is important because to ascertain the source of a message and to ensure that it has not come from an intruder.

Nonce: a value used oNly-Once, and important for 'freshness' of unique values between the node being interrogated, and the node verifying the authenticity of the newcomer.

Blacklist: the bank of recently-declined public keys identifying attempted intrusions. This list should be cluster-specific, ephemeral, and hosted by a General for a limited time.

Shadows: shares in a symmetric key between two or more ranked members. This is to protect against granting authority without group consent.

XOR: representing 'exclusive or', or the logical operation that is true only if its arguments differ, so that one is true and the other is false. Used to create shadows of shared secrets. Timestamps: allow the authorisation model to determine the most appropriate replacement for lost entities by the length of time the others have served the system for.

B. Privacy

Where the GDPR expresses the notion of 'necessary processing', the network should refuse any request for data access beyond the minimum required for processing. Anonymisation, pseudonymisation and tokenisation are explored as lightweight alternatives to the heavier and more complex security encryption ciphers for enforcing consent management and access control [42].

Anonymisation is the process of removing personal identifiers that may lead to the identification of an individual. Now, there are no named individuals in the authorisation system, or the aquaponics smart-farm. However, it could be seen as wise, wherever possible, to disguise the identity of higher-ranking devices able to grant authority as they will undoubtedly attract attention. For this reason, in a cluster of several devices, the MAC addresses do not broadcast BLE information to join the network - they do not actively operate GATT server-client advertising. The ranked devices also pose as any other regular system, and they cannot be communicated with by an unauthorised device.

Pseudonymisation is to replace any information which could be used to identify an individual, with a pseudonym. Again, although there are no individuals in the application, pseudonymisation is a useful way of quickly locating devices by their existing security and communications attributes. By using the MAC address as a location and crop reference that will only make sense to the farm operators, it provides an easy way of managing systems without creating additional data.

Tokenisation is an advanced form of pseudonymisation, whereby a meaningful piece of data such as an account number is changed into a seemingly random number which has no value if breached. Tokens are useful to safely send over public channels, disguising their true value. In the authorisation design, tokens are used in the form of nonces for freshness to verify authenticity on a single-use basis.

C. Autonomy

Self-healing and automated management in the authorisation tiers are the objective for an IoT consensus mechanism capable of operating remotely. Policy or attribute-oriented authorisation systems have demonstrated good functionality using 'if' and 'or' logic. Policies can form the fundamentals of security models; write, execute permissions, replacing old data with new, and providing workflow. There are mature and well-known security models used in both military and civilian applications which focus on data confidentiality and integrity. These models are mature and operate on effective simple rule sets making them a suitable influence on IoT-oriented and constrained environments. Notably, these models are Take-Grant [43], depicting transference of rights, Biba [44], for preservation of data integrity, and Multi-Level Security (MLS) [45], providing classification. Take-Grant represents a directed graph, in which vertices are either subjects or objects, and the edges between them depict the rights towards the destination. There are two possible rights which occur in every instance, take and grant, forming four rules; subject takes rights of an object, subject grants own rights to another object, subject creates a new object, or subject removes rights it has over an object. This model is utilised when promoting devices in the event of a lost, existing one. The Biba integrity model

is characterised by the phrase “read up, write down”, and defines the rule that a subject of a given integrity level must not read data at a lower integrity level, nor write to a higher one. This model influences the protection of higher-ranked devices against lower-ranked. MLS demonstrates processing data between classifications of devices. Applied to the design ‘ranks’, and there is a hierarchy to prevent users from obtaining access to data for which they lack authorisation. Now, self-protection can be difficult to automate, and there are no strict guidelines on how to do this - so the design has formed a three-tier proposal by which sets of ‘clustered’ devices have limited authority. When one of those authoritarian devices falls, it is replaced by the most mature unranked device. This concept aims to maintain a limited set of backup whilst employing, but not trusting, any other - when group consensus has been obtained, of course.

D. Consent

Group consent has been exercised in the field of cryptocurrency with enormous success. The same processes for validation influence the design proposal, but preferably without the cumulative data chains that accompany them. Attaining group consensus is far preferable to multi-party authentication, since secret sharing using eXclusive-or (XOR) [46], functions between a few devices uses considerably less energy consumption than escrow, particularly the X.509 infrastructure, Zero Knowledge Succinct Non-Interactive Argument of Knowledge (ZK-SNARKS) [47], or centralised trusted third parties such as Key Generation Centres (KGC) [48]. Where data validation is concerned, mining should be discouraged. In many cryptocurrency applications, mining exemplifies equality and robustness using the Proof of Work (PoW) mechanism. However, a domain in which clusters are contained within finite bounds of authority and geographic range, ownership is not such a concern, and Proof of Stake (PoS), can provide a more suitable template. The great disadvantage associated with PoS is ‘blockchain oligarchy’, but this is actually should be maintained. Many other consensus mechanisms exist of course; Proof of Elapsed Time (PoET), Practical Byzantine Fault Tolerance (PBFT), Proof of Importance (PoI), Proof of Capacity (PoC), Proof of Authority (PoA), Proof of Experience (PoE), as common examples. The objective of this exercise is to simplify and automate authorisation towards the lowest consumptions in time, power and energy available to this specific IoT application. This is possible by reducing processes, maths, and avoiding complex procedures such as learning algorithms or reputations systems.

V. Mechanism secrecy

This section presents the formal verification methods, design and test results of the authorisation model using AVISPA.

A. Formal verification

The Automated Validation of Internet Security Protocols and Applications (AVISPA) platform was used in conjunction with the Security Protocol ANimator (SPAN) tool for formal verification of the algorithms described in the activity diagrams. The aim was to ensure secrecy in certain attributes, whilst using as few security functions as possible in exchange for identifiers or privacy functions.

The Dalov-Yao threat model was used to test the secrecy and freshness of values sent between devices, guarding against eavesdroppers and intruders seeking to gain knowledge over a deliberately insecure channel. The protocol was written in High Level Security Programming Language (HLSPL), and loaded into the SPAN interface within AVISPA. SPAN provides four backends for verification:

OFMC: [49], On-the-Fly Model Checking performs protocol falsification and bounded session verification. OFMC is demand-driven and uses a number of symbolic, constraint-based techniques such as the lazy intruder, and constraint differentiation, for typed and untyped protocol models.

CL-ATSE: [50], Constraint-Logic-based ATtack SEarcher takes a protocol input using the AVISPA compiler language Intermediate Format (IF), and models all reachable states to determine whether an attack is possible under the Dolev-Yao intruder model. This is state-based security modelling, such as fairness, freshness, authentication and secrecy, including the XOR and exponentiation operators.

SATMC: [51], SAT-based Model-Checker considers typed protocol models, providing falsification and bounded session verification using a SAT solver. This is a satisfiability solver which takes a Boolean input, and outputs results based on whether the variables can confirm it is true. SATMC are not ideally used for trees.

TA4SP: [52], Tree-Automata-based Protocol Analyser performs unbounded verification by approximating intruder knowledge using tree languages as opposed to strings, based on tree automata [53]. Secrecy properties in the model can be revealed as flawed by under-approximation, or safe throughout any number of sessions by over-approximation.

SPAN also provides three types of animation to simulate the protocol itself, the methods of an intruder, and a successful attack. If the proposed authorisation model is determined as safe by all, if not most the verification backends, then an attack simulation will not be possible.

B. Algorithm overview

Below is an overview of the authorisation algorithm. It describes any device joining the network, from zero members to unlimited. From here, the integrity of the cluster will be attested by the number of ranked devices being checked. If there are not enough ranked devices at each of the three hierarchy tiers, the algorithm will automatically add them by promotion to a rank as devices accumulate to the network. When the ranks are full, they will remain in this state until clusters begin to realise beyond range, or when a device fails.

C. Function Symbols

1) N is the identifier of the device which a message is destined for, towards, or containing identifier material for challenge-response of authenticity. This identifier

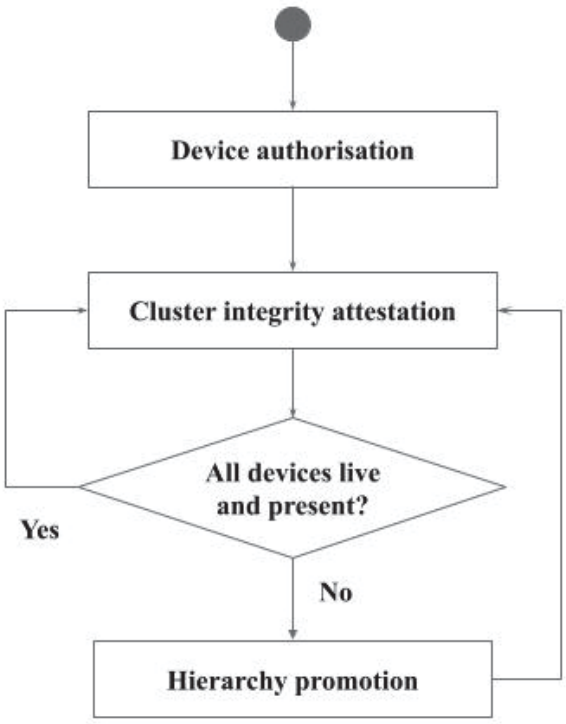

Fig. 1. Authorisation Life Cycle

TABLE II. FUNCTION SYMBOLS

Ref	Symbol	Meaning
1	N	Identifier of device N, annotates direction of message
2	MAC-n	Media Access Card (MAC) address of device
3	Pub-n	Public key of Device n, providing decryption for only device n
4	Priv-n	Private key of device n, providing signature proving origin from device n
5	Nn	Nonce from device n, showing freshness of value
6	s	Secret, unique time-limited password for network entry
7	I	Instructions, code for new aquaponics system
8	Sn	Secret shadow of device n, their part of the whole rank hash
9	$\oplus$	XOR function, eXclusive OR to create secret shadows
10	R	Rank, either Field Marshal, General, or Unranked
11	Ts	Timestamp, the time marking to prove age of device for trustworthiness
12	RS	Random String, a bitstring provided to create shadows from a bigger array
13	RH	Rank Hash, the full combined rank secret from the collective shadows

could be a public key, a signature, MAC address, or time-limited password. It is not really important in practice what this data is, but serves useful in the activity diagrams as to ascertain workflow.

2) MAC-n represents a pseudonymised identifier of all devices in the network, of 12 characters, separated by colons 'xx:xx:xx:xx:xx:xx'. Although MAC addresses can be spoofed, they are helpful when scanning clusters to ascertain locations and components, as the 12 characters can be used as legible abbreviations for location, compass direction, crop type, age, and so forth. A MAC address reading "SY:SH:SW:14:W1:21" could be interpreted as "South Yorkshire, Sheffield, South West, system 14, Wasabi plant 1, year 2021" for example. MAC addresses allow pseudonymisation so that a separate reference table to address meaningless identifiers is not needed. For protection against spoofing, all messages are encrypted with corresponding public keys; even though a message may be sent to an impersonator, they will be unable to decrypt it.
3) Pub-n or the public key of the public and private key pair belonging to device n, is the key designed to be in the public domain. All devices have a public key, but some devices can be used to communicate with freely, and others have restrictions. This is to prevent ranked devices from reading, writing, or executing commands provided by untrusted guests on the network. This key is also used as an identifier, legible to machines more than humans.
4) Priv-n or private key of the public and private key pair belonging to device n, is the key used for proving authenticity of messages sent from one device to another. Within the activity diagrams, this key denotes a Digital Signature (DS), an algorithm representing part of a device's private key that proves a message has come from the device it claims to have come from. This is an imperative piece of data to present imposters from communicating with authorised members.
5) Nn or a oNly-once, nonce, value sent from one device to another as part of a challenge-response scenario. Now, when dealing with authentication or unique messages, a nonce represents freshness; a random piece of data transferred within a limited period of time that cannot be stored and replicated to falsely impersonate during time to come. The nonce can be used more than once within a challenge-response scenario if the model does not challenge the nonce's freshness in favour of something else, such as a session secret. If it is the nonce under adversarial challenge, then it takes priority of freshness, and should only be used once.
6) s is the secret, and the freshness priority of the model's challenge-response adversary for authorising the new device. This secret represents a piece of data given to the device when it instigates authorisation to the network, and is challenged by the existing ranks. It could be a simple phrase or word, or a random bitstring,

for example 'Fish-Division-66'. It must be ephemeral, limited by time, and not used amongst other devices. This is the data that will be criticised the most during authentication protocol checking, and is the secrecy priority.

7) I is the instructions set which will be sent to the new device following authentication of identity, and authorisation as a network member to fulfill sensor-actuator readings of its own system.
8) Sn is the shadow of device n, or the part secret it holds as a member of a rank. The secret shadows together form a Rank Hash (RH), a bitstring representing the complete symmetric key to all the devices in a cluster of the same hierarchy position and cluster when each shadow is XOR'd against the Random String (RS).
9) $\oplus$ eXclusive OR, or XOR function, used to create ciphertext from the Boolean logic of 1 or 0 input to create a different output in the same manner as a truth table.
10) R is the device Rank, of which there are limited numbers of Field Marshals (FM), Generals (G), and unlimited Unranked (U) devices in each cluster. The limited authority sets provide finite storage limits, chains of data which cannot exceed certain lengths so that the chains within clusters cannot overwhelm the capacity of constrained devices. The ranks serve only to separate permissions far away from the acquisition of the public, so that intruders are limited to only those who undertake business continuity measures as an event of extreme loss - which under a self-healing network, should not happen.
11) Ts Timestamps are the essence of the model's 'experience' algorithm. Where a ranked device fails its part to complete cluster attestation, the algorithm will seek to promote the oldest unranked device to that required position. Of course other measures can be added to this procedure, such as checking the sensor reading history of the unranked device to check it is actually doing what it is supposed to do, but the theory is that the longer the device has been there, the more valid it becomes.
12) RS Random Strings are required as part of the XOR functioning to further obfuscate the RH secret, split into shadows. Processed on a single device and passed onto other devices using encryption, this becomes impossible to derive by an intruder.
13) RH or Rank Hash, is the shared symmetric key between all devices of a rank, accessible only by a Field Marshal or upon cluster attestation when a single device temporarily acts as a central authority by which to request and compare the shadows of all others in that cluster rank. In doing so the rank will either have a matching RH and be complete, or it will not match, and a new device will reset the rank's hash and shadows.

D. Device authorisation (Algorithm 1)

The activity diagram below describes how an unauthorised device attains authorised, unranked status. The unauthorised device will instigate the process by detecting an unranked, authorised device, and approach it using the accessibility of its public key. Device A should be made obvious and approachable to new devices, perhaps coupled with the advertising beacon of a BLE GATT stack, since that is what the MAC address of an ESP32 can source within cluster range.

Device B will send their MAC address and public key, signed by the private key so that A can authenticate the contents and origin using the signature. Device A will then respond with a challenge. This is the first of several authentication steps that will be formally verified later on. Device B receives the challenge, containing identifiers of A and B, A's public key for returning the set, a nonce from A, encrypted using the public key that B previously supplied, and signed by A's private key so that B can check that the public and private keys match. B will return these values with a secret, s, such as the aforementioned 'Fish-Division-66' password, or other suitably fresh piece of information proving that B is applying for the right cluster, within the finite time. The value 's' is the freshness, and will expire following authorisation of the device. If the secrecy following verification is found to not be completely fresh, this indicates a failed authorisation and the device will be blacklisted by both MAC and its public key. Blacklists last only for a limited time to allow for investigation and are stored by a cluster General.

Device A will verify the contents, dispose of the nonce and respond with a set of instructions containing sensor reading code used for a functioning aquaponics system.

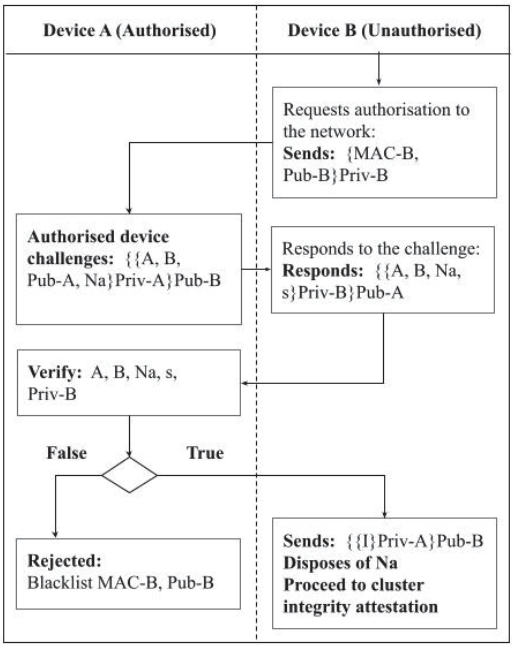

Fig. 2. Device Authorisation

1) Results for the goals of secrecy of S:

- OFMC: summarised as safe under a bounded number of sessions.

- CL-ATSE: summarised as safe under bounded number of sessions and typed model.
- SATMC: summary inconclusive.
- TA4SP: summarised as safe under typed model, over approximation and unbounded number of sessions, no attack trace found.

2) Results for the goals of secrecy of I:

- OFMC: summarised as safe under a bounded number of sessions.
- CL-ATSE: summarised as safe under bounded number of sessions and typed model.
- SATMC: summary inconclusive.
- TA4SP: summarised as safe under typed model, over approximation and unbounded number of sessions, no attack trace found.

In both secrecy instances, three of four backend verifiers considered the model safe, but the sat-solver SATMC could not conclude the secrecy - this was most likely due to the fact there were too few messages containing the secret information, which is a positive thing for freshness.

E. Cluster integrity attestation (Algorithm 2)

When there is a system check to verify the presence and validity of known devices, this is known as integrity attestation. Applied to ranked sets of Field Marshals and Generals within a given cluster, this is cluster integrity attestation, and is performed every time a device joins the network, or a device is found unresponsive, and only applies to authorised devices.

Device A broadcasts a request for rank shadows by sending their public key, rank, and a nonce, encrypted by the public key of each device, to all the MAC addresses it has stored as rank members. This algorithm will take place on two ranks, Field Marshals and Generals. The corresponding ranks will reply with their recorded rank, their shadow, the same nonce as received, signed individually, and encrypted by the public key provided by device A. The secrecy aspect here is the shadow, and should only be sent once.

Device A will then verify the rank and nonce, and XOR all the shadows including its own to verify the rank hash, which is stored with the Field Marshal. The separation between ranks ensures that a rank secret cannot be gained by an intruder by any single General. If the shadows match the rank hash, the cluster's integrity is attested, but if not, then a new device must be promoted to a new rank in a separate algorithm.

1) Results for the goals of secrecy of shadows Sb...Sn:

- OFMC: summarised as safe under a bounded number of sessions.
- CL-ATSE: summarised as safe under bounded number of sessions and typed model.
- SATMC: summarised as safe under strongly typed model, bounded number of sessions, and bounded message depth. No attack traces were found.
- TA4SP: summarised as safe under typed model, over approximation and unbounded number of sessions, no attack trace found.

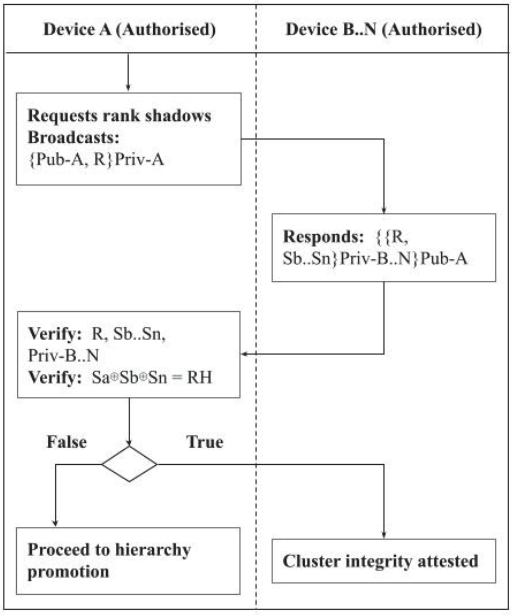

Fig. 3. Cluster Integrity Attestation

F. Hierarchy promotion (Algorithm 3)

Hierarchy promotion tests the age or 'experience' of devices, not dissimilar to a reputation system, but in a really simple way. The algorithm seeks the oldest timestamp from authorised but unranked devices in order to complete a FM or General cluster, to self-heal the backup system of the authorisation system. Of course, this only applies to authorised devices. One device will temporarily assume the position of a centrally trusted device and perform all the calculations and broadcasting. This should be the oldest device in the cluster.

Device A broadcasts a request for all devices belonging to its own rank, with a public key, the rank, and a new unique random bitstring for a nonce, signed with its private key. Ranked devices of the same cluster will be registered with the central device by their MAC address and timestamp. In this instance, the timestamp acts as a piece of secrecy to discourage responses from intruders - responses from imposter MAC addresses will not match the recorded timestamp. The central device will also detect the unranked MAC with the oldest timestamp. Since unranked devices are designed to be approachable and accessible, broadcasting the same message across the whole cluster will attract a prompt and suitable device.

Devices B through N will respond, depending on how many devices the rank is lacking. They will either attest the rank they currently hold, or show as unranked but of oldest age(s) of the cluster. Device A will verify the corresponding public and private keys with ranks, the original nonce and the fresh timestamp value for unranked devices (timestamps belonging to ranked devices will match device A's records).

From here, a random bitstring is generated to XOR a new set of secret shadows for each device, which are subsequently

sent to each device to update their shadow. Previous shadows are discarded and so data chains remain finite. Finally, the full rank hash secret is forwarded to the Field Marshals, whose only purpose is to maintain the symmetric keys, or full rank hashes, of each cluster, and as many clusters as possible.

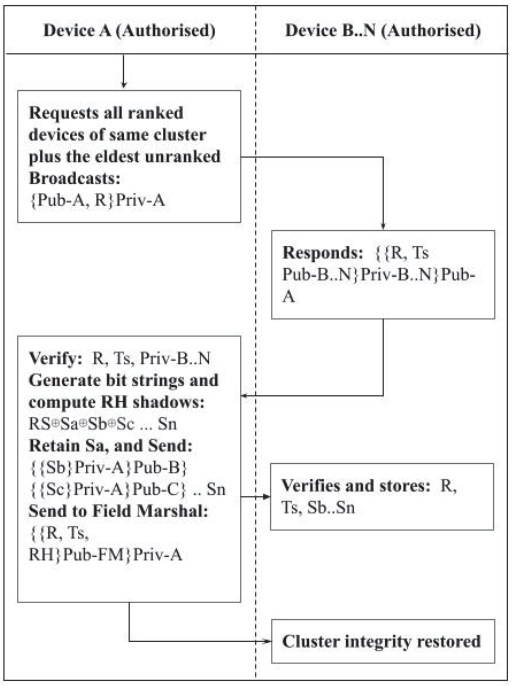

Fig. 4. Hierarchy Promotion

1) Results for the goals of secrecy of Ts:

- OFMC: summarised as safe under a bounded number of sessions.
- CL-ATSE: summarised as safe under bounded number of sessions and typed model.
- SATMC: summarised as safe under strongly typed model, bounded number of sessions, bounded message depth.
- TA4SP: summarised as safe under typed model, over approximation and unbounded number of sessions, no attack trace found.

2) Results for the goals of secrecy of Sb...Sn:

- OFMC: summarised as safe under a bounded number of sessions.
- CL-ATSE: summarised as safe under bounded number of sessions and typed model.
- SATMC: summarised as inconclusive.
- TA4SP: summarised as safe under typed model, over approximation and unbounded number of sessions, no attack trace found.

In the second instance, for testing the secrecy of shadows Sb..Sn, the sat-solver SATMC was inconclusive - this was most likely due to the fact there were too few messages containing the secret information, which is a positive thing for freshness.

VI. Conclusion and Further Work

Through formal verification, the authorisation model has been proven as safe, through a majority of three out of four model checking methods, and in some instances, four out of four. The sat-prover was sometimes inconclusive, probably due to the efficiencies of the messaging between nodes - but these efficiencies were positive towards a lightweight effort. This section concludes the benefits of the authorisation model:

A. Low consumptions

By reducing the number of messages between nodes to as few as possible, the freshness of the nonces and secrecy of the priority data is assured. This has been proven using the four backend model checkers in the AVISPA -SPAN framework. In addition, the less processing that the ESP32 has to do for ascertaining authenticity for each new or refreshed network member, the less energy the network will use, and the longer the battery life of devices will last.

B. Self-healing

The design includes a periodic cluster integrity attestation function to assure the network that the hierarchy devices are still live and responsive. The primary function of the hierarchy is to separate levels of data access away from access to the public, or authorised devices allowed to participate on the network that may not be as trustworthy as they should. In comparison to TLS which is either permitted access or not, this tiered system separates rights into the four levels between public, unranked, high-ranked, and secret backup. In addition to preventing malicious attempts, this system is designed to form automated business continuity in the event of network loss.

C. Limited chain

The restricted nature of the privileged devices within each cluster means that only the correlating set of identifiers, secret shadows and keys are maintained in a type of chain. The repetition of this data is limited to only a few devices, and although there will be an overlap of the higher rank, Field Marshal, between clusters due to the longer range, it is predictable, finite, and ephemeral by nature - when a previous device is lost, the new data set will replace their part in the chain. An ephemeral application enables the robustness of data replication and dispersion in the same enabling way as the famous blockchain, without the unmanageable scalability issues of storage.

D. Agnostic application

The security properties expected of any desktop, mobile or hardware application require the same as TLS - a public and private key pair, hash functions, Boolean logic (XOR), exponentiation operators, hash, and symmetric cryptography. These functions are native to the ESP32 hardware acceleration, and since security is becoming an increasingly important facet to any application, a multi-purpose approach to functions between applications lessens the burden of processing and

storage. Cryptographic functions have therefore been used for identification and other attributes to reduce data volume.

E. Consensus

Consent is becoming more important for IoT applications because of privacy, and security can enable the principles of the GDPR by supporting privacy through consent. Although the aquaponics smart-farm does not utilise the private and sensitive information of living subjects, it exemplifies the utility of a tiered authorisation system. The separation of privilege levels, pseudonymisation of device identifiers, and anonymity of 'rankings' demonstrates the overlap between security and privacy using majority consensus and protection of authorisation.

F. Further work

This research demonstrated an automated application of self-healing and ephemeral data sets, replicated in a limited way between closely associated devices. Applications of such a managed authorisation system could extend into domains such as healthcare, domestic and industrial. Such an energy and storage-conscious design could greatly benefit challenges faced in energy production, environmental conservation and healthcare, in which monitoring over long periods of time could take advantage of smaller data chains and device restoration.

REFERENCES

[1] Y. C. P. Kirci and E. Ozturk, "Smart greenhouse and smart agriculture," https://www.fruct.org/publications/acm29/files/Kir.pdf, accessed: 2021-12-8.

[2] A. R. Yanes, P. Martinez, and R. Ahmad, "Towards automated aquaponics: A review on monitoring, IoT, and smart systems," *J. Clean. Prod.*, vol. 263, p. 121571, Aug. 2020.

[3] L. Axon, "Privacy-awareness in blockchain-based PKI," *CDT Technical Paper*, 2015.

[4] S. Kim, Y. Kwon, and S. Cho, "A survey of scalability solutions on blockchain," in *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, Oct. 2018, pp. 1204–1207.

[5] J. Truby, "Decarbonizing bitcoin: Law and policy choices for reducing the energy consumption of blockchain technologies and digital currencies," *Energy Research & Social Science*, vol. 44, pp. 399–410, Oct. 2018.

[6] J. A. Ali, Q. Nasir, and F. T. Dweiri, "Business continuity framework for internet of things (IoT) services," *International Journal of System Assurance Engineering and Management*, vol. 11, no. 6, pp. 1380–1394, Dec. 2020.

[7] M. Kolomeets and A. Chechulin, "Analysis of the malicious bots market," in *2021 29th Conference of Open Innovations Association (FRUCT)*, May 2021, pp. 199–205.

[8] M. Gorbunova, P. Masek, M. Komarov, and A. Ometov, "Distributed ledger technology: State-of-the-art and current challenges," *Comput. Sci. Inf. Syst.*, no. 00, pp. 37–37, 2021.

[9] I. Struchkov, A. Lukashin, B. Kuznetsov, I. Mikhalev, and Z. Mandrusova, "Agent-Based modeling of blockchain decentralized financial protocols," in *2021 29th Conference of Open Innovations Association (FRUCT)*, May 2021, pp. 337–343.

[10] M. Bulygin and D. Namiot, "A new approach to clustering districts and connections between them based on cellular operator data," in *2021 29th Conference of Open Innovations Association (FRUCT)*, May 2021, pp. 71–80.

[11] S. Leech, D. Malone, and J. Dunne, "Heads or tails: A framework to model supply chain heterogeneous messages," in *2021 30th Conference of Open Innovations Association FRUCT*, Oct. 2021, pp. 129–140.

[12] A. J. Goldsmith and S. B. Wicker, "Design challenges for energy-constrained ad hoc wireless networks," *IEEE Wirel. Commun.*, vol. 9, no. 4, pp. 8–27, Aug. 2002.

[13] S. Alnefaie, S. Alshehri, and A. Cherif, "A survey on access control in IoT: models, architectures and research opportunities," *Int. J. Secur. Netw.*, vol. 16, no. 1, pp. 60–76, Jan. 2021.

[14] S. Pal, A. Dorri, and R. Jurdak, "Blockchain for IoT access control: Recent trends and future research directions," *arXiv*, Jun. 2021.

[15] P. Kanakaraja and E. Al, "IoT enabled BLE and LoRa based indoor localization without GPS," *TURCOMAT*, vol. 12, no. 4, pp. 1637–1651, Apr. 2021.

[16] L. Vishwakarma and D. Das, "SCAB - IoTA: Secure communication and authentication for IoT applications using blockchain," *J. Parallel Distrib. Comput.*, vol. 154, pp. 94–105, Aug. 2021.

[17] O. Green, "HashGraph—Scalable hash tables using a sparse graph data structure," *ACM Trans. Parallel Comput.*, vol. 8, no. 2, pp. 1–17, Jul. 2021.

[18] I. A. Prostov, S. S. Amfiteatrova, and N. G. Butakova, "Construction and security analysis of private directed acyclic graph based systems for internet of things," in *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*. ieeexplore.ieee.org, Jan. 2021, pp. 2394–2398.

[19] S. Rautmare and D. M. Bhalerao, "MySQL and NoSQL database comparison for IoT application," in *2016 IEEE International Conference on Advances in Computer Applications (ICACA)*. ieeexplore.ieee.org, Oct. 2016, pp. 235–238.

[20] J. Bhogal and I. Choksi, "Handling big data using NoSQL," in *2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops*. ieeexplore.ieee.org, Mar. 2015, pp. 393–398.

[21] J. L. Hernández-Ramos, A. J. Jara, L. Marin, and A. F. Skarmeta, "Distributed capability-based access control for the internet of things," *Journal of Internet Services and Information Security (JISIS)*, vol. 3, no. 3/4, pp. 1–16, 2013.

[22] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed access control in clouds," in *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*, Nov. 2011, pp. 91–98.

[23] D. He, J. Bu, S. Zhu, S. Chan, and C. Chen, "Distributed access control with privacy support in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 10, pp. 3472–3481, Oct. 2011.

[24] F. Kausar, M. A. K. Sadiq, and H. M. Asif, "Convergence of blockchain in IoT applications for heterogeneous networks," in *Real-Time Intelligence for Heterogeneous Networks: Applications, Challenges, and Scenarios in IoT HetNets*, F. Al-Turjman, Ed. Cham: Springer International Publishing, 2021, pp. 71–86.

[25] R. Arshad, S. Zahoor, M. A. Shah, A. Wahid, and H. Yu, "Green IoT: An investigation on energy saving practices for 2020 and beyond," *IEEE Access*, vol. 5, pp. 15 667–15 681, 2017.

[26] H. Tschofenig and T. Fossati, "Transport layer security (tls)/datagram transport layer security (dtls) profiles for the internet of things," in *RFC 7925*. Internet Engineering Task Force, 2016.

[27] E. P. Frigieri, D. Mazzer, and L. Parreira, "M2m protocols for constrained environments in the context of iot: A comparison of approaches," in *International Telecommunications Symposium*. researchgate.net, 2015, p. 5.

[28] J. Ahamed and F. Khan, "An enhanced context-aware capability-based access control model for the internet of things in healthcare," in *2019 Sixth HCT Information Technology Trends (ITT)*, Nov. 2019, pp. 126–131.

[29] W. Newhouse, M. Ekstrom, J. Finke, and M. Harriston, "Securing property management systems," National Institute of Standards and Technology, Tech. Rep., Mar. 2021.

[30] A. Calabrò, S. Daoudagh, and E. Marchetti, "Integrating access control and business process for GDPR compliance: A preliminary study," in *ITASEC*, 2019.

[31] L. Grassi, D. Khovratovich, C. Rechberger, A. Roy, and M. Schofnegger, "Poseidon: A new hash function for zero-knowledge proof systems," in *30th {USENIX} Security Symposium ({USENIX} Security 21)*. usenix.org, 2021.

[32] B. B. Rao and A. A. Waoo, "DESIGN a NOVEL APPROACH FOR TOKEN BASED AUTHENTICATION IN IOT NETWORKS," *Ilkogretim Online*, vol. 20, no. 4, 2021.

[33] R. Bharanidharan, "A novel blockchain approach for improve the performance of network security using polynomial ephemeral Blockchain-Based secure routing in wireless sensor network," *J. Comput. Theor. Nanosci.*, vol. 17, no. 12, pp. 5598–5604, 2020.

[34] A. D. Miyazaki and A. Fernandez, "Consumer perceptions of privacy and security risks for online shopping," *J. Consum. Aff.*, vol. 35, no. 1, pp. 27–44, Jun. 2001.

[35] H. Blaine, "The threat landscape of PKI: System and cryptographic security of x. 509, algorithms, and their implementations," *NATO Sci. Ser. Ser. A. Life Sci.*, p. 286, 2013.

[36] D. Hopwood, S. Bowe, T. Hornby, and N. Wilcox, "Zcash protocol specification," *GitHub: San Francisco, CA, USA*, 2016.

[37] R. K. Raman and L. R. Varshney, "Dynamic distributed storage for blockchains," in *2018 IEEE International Symposium on Information Theory (ISIT)*. ieeexplore.ieee.org, Jun. 2018, pp. 2619–2623.

[38] J. Roth, F. Schär, and A. Schöpfer, "The tokenization of assets: Using blockchains for equity crowdfunding," Aug. 2019, accessed: 2021-12-3.

[39] H. Treiblmaier and C. Sillaber, "The impact of blockchain on e-commerce: a framework for salient research topics," *Electron. Commer. Res. Appl.*, vol. 48, p. 101054, 2021.

[40] K. Kypriotaki, E. Zamani, and G. Giaglis, "From bitcoin to decentralized autonomous corporations-extending the application scope of decentralized peer-to-peer networks and blockchains," in *International conference on enterprise information systems*, vol. 2. scitepress.org, 2015, pp. 284–290.

[41] A. Bates, B. Mood, M. Valafar, and K. Butler, "Towards secure provenance-based access control in cloud environments," in *Proceedings of the third ACM conference on Data and application security and privacy*, ser. CODASPY '13. New York, NY, USA: Association for Computing Machinery, Feb. 2013, pp. 277–284.

[42] S. Daoudagh, E. Marchetti, V. Savarino, R. Di Bernardo, and M. Alessi, "How to improve the GDPR compliance through consent management and access control," in *ICISSP*, 2021, pp. 534–541.

[43] J. Frank and M. Bishop, "Extending the take-grant protection system," https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.51.907&rep=rep1&type=pdf, accessed: 2021-9-15.

[44] K. J. Biba, "Integrity considerations for secure computer systems," MITRE CORP BEDFORD MA, Tech. Rep., 1977.

[45] B. Danner, C. Muckenhirn, T. Valle, C. McElveen, J. Bragdon-Handfield, and A. Colegrove, "Multilevel security feasibility in the M&S training environment," in *Proceedings of the Interservice/Industry Training Simulation and Education Conference (I/ITSEC)*. Citeseer, 2002.

[46] N. Ferguson and B. Schneier, *Practical cryptography*. Wiley, 2003.

[47] A. M. Pinto, "An introduction to the use of zk-SNARKs in blockchains," in *Mathematical Research for Blockchain Economy*. Cham: Springer International Publishing, 2020, pp. 233–249.

[48] M. Shen, H. Liu, L. Zhu, K. Xu, H. Yu, X. Du, and M. Guizani, "Blockchain-Assisted secure device authentication for Cross-Domain industrial IoT," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 942–954, May 2020.

[49] S. Mödersheim, L. Viganò, and D. Basin, "Constraint differentiation: Search-space reduction for the constraint-based analysis of security protocols," *J. Comput. Secur.*, vol. 18, no. 4, pp. 575–618, Jun. 2010.

[50] M. Turuani, "The CL-Atse protocol analyser," in *Term Rewriting and Applications*. Springer Berlin Heidelberg, 2006, pp. 277–286.

[51] A. Armando, R. Carbone, and L. Compagna, "SATMC: A SAT-Based model checker for Security-Critical systems," in *Tools and Algorithms for the Construction and Analysis of Systems*. Springer Berlin Heidelberg, 2014, pp. 31–45.

[52] N. Liu, W.-Y. Zhu, and Y.-F. Zhu, "Security protocol analysis based on rewriting approximation," in *2009 Second International Symposium on Electronic Commerce and Security*, vol. 1. ieeexplore.ieee.org, May 2009, pp. 318–322.

[53] E. A. Emerson and C. S. Jutla, "Tree automata, mu-calculus and determinacy," in *FoCS*, vol. 91. Citeseer, 1991, pp. 368–377.