Dissertation Thesis

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Computer Science

Context-aware security of Internet of Things applications

Michal Trnka

Supervisor: doc. Ing. Karel Richta, CSc.
Supervisor–specialist: Ing. Tomáš Černý, MSc., Ph.D.
Field of study: Electrical Engineering and Information Technology
Subfield: Information Science and Computer Engineering
March 2021

Acknowledgements

This thesis would not be possible without the help of many people and institutions. Two individuals provided me crucial support during my research. My supervisor doc. Richta was supporting me and leading me through the challenging process of doctoral studies. Co-supervisor dr. Černý was always around for advice, feedback, or consultation.

My sincere thanks go to my family. My parents were supporting me during the whole education process. Moreover, I would like to mention my wife, who had enough patience, understanding and moved with me to the USA for nine months with a newborn baby. Furthermore, final thoughts go to my two daughters that often had to spend evenings or even weekends without their beloved father.

I want to thank Czech Technical University in Prague for providing me education comparable to a university anywhere on the planet. Next, I want to express my sincere gratitude to the Fulbright committee for sponsoring my visiting research and to Baylor University for allowing me to do a crucial part of my research in their facilities in Waco, TX.

Declaration

I declare that this thesis has been composed solely by myself. I have used ideas and content of my own properly published articles. Except where states otherwise by reference or acknowledgment, the work presented is entirely my own.

Prague, March 3, 2021

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškerou literaturu na které můj výzkum staví. V práci jsem vycházel z myšlenek a obsahu svých řádně publikovaných prací.

V Praze, 3. března 2021

Abstract

Security is one of the major research topics within the Internet of Things landscape. Recently, the development of Internet of Things solutions has significantly moved forward. However, security is lacking behind all the notable progress that has been accomplished.

In my research, I mainly focus on authentication, authorization, and partially on identity management of Internet of Things network participants. I consider the challenges from the software engineering perspective - architecture and high-level design of the authentication or authorization solutions. It also implies that I operate on the application layer of the networking stack. Specifically, I concentrate on three main areas - context retrieval, context-aware authorization, and identity and security rules sharing in the scope of the Internet of Things. I evaluate the current state of the art to give an overview of where my research stands compared to existing research.

Initially, I propose a method to determine context from the network neighborhood. The method evaluates available devices on the network and tracks their temporal changes. The changes in the composition of the devices on the network are quantified and used as additional contextual information.

The core part of the research is focused on authorization. I describe a context-aware extension of Role-Based Access Control using security levels. Levels are linear single value representation of the context. The user's level is determined during logging into the application via various configurable context resolvers.

The last part of the thesis covers identity management in the Internet of Things. I utilize a centralized element to store devices' and users' identities and provide authentication in the form of a token. On top of that, the central server provides additional attributes, like roles in the token.

Keywords: Internet of things, dissertation, software security, software engineering, authentication, authorization

Supervisor: doc. Ing. Karel Richta, CSc.
Department of Computer Science
Resslova 9, E-430
Prague

Abstrakt

Zabezpečení je jedna z klíčových oblastí výzkumu v oblasti Internetu věcí. V nedávné době se vývoj Internetu věcí významně posunul kupředu, nicméně zabezpečení stále zůstává pozadu za pokrokem, který byl dosažen ve zbylých oblastech.

Ve své práci se zaměřuji především na autentizaci, autorizaci a částečně na správu identit účastníků komunikace Internetu věcí. A na tuto problematiku nahlížím z perspektivy softwarového inženýrství, tj. zajímám se o architekturu a obecnou strukturu řešení. Znamená to tedy, že operuji na aplikační vrstvě síťového modelu. V popředí mého zájmu stojí tři oblasti, kterými jsou získání kontextu, autorizace s ohledem na kontext, a sdílení identit a zabezpečovacích pravidel v rámci Internetu věcí. V této práci shrnuji rešerši současného stavu poznání a popisuji kam patří mé bádání v rámci stávajícího širšího výzkumu.

Zaměřil jsem se nejprve na metodu zjišťování kontextu ze síťového okolí, která rozpoznává dostupná zařízení v síti, a vyhodnocuje jejich vývoj v průběhu času. Změny ve složení těchto zařízení jsou poté kvantifikovány a požity jako další kontextová informace.

Nejdůležitější část mého výzkumu zaujímá autorizace nebo-li ověření přístupových oprávnění. V této části rozšiřuji zabezpečení pomocí rolí o kontextový element v podobě úrovně zabezpečení. Tzn., že daná úroveň je lineární hodnota reprezentující stávající kontext, a úroveň zabezpečení uživatele je vyhodnocena během jeho přihlášení do aplikace pomocí různých nastavitelných rozhodovacích mechanismů, které vyhodnocují specifické aspekty kontextu.

V poslední části mé práce se zabývám správou identity na Internetu věcí. K tomu využívám centrální prvek pro ukládání identity zařízení a uživatelů. Tento prvek vydává token, který je používán k přihlašování do síťového prostředí. Může však obsahovat i další atributy pro autorizaci.

Klíčová slova: Internet věcí, dizertace, aplikační zabezpečení, softwarové inženýrství, autentikace, autorizace

Překlad názvu: Zabezpeční systémů pro Internet věcí s ohledem na kontext

Contents

Chapter 1

Introduction

Internet of Things (IoT) is an environment in which numerous heterogeneous and possibly small devices interact and cooperate. Each device might have a specialized function where the overall ecosystem provides various and possibly more complex features. Currently, IoT solutions are deployed in diverse domains that range from agriculture through transportation, retail, physical security, industrial automation, home solutions, healthcare all the way up to defense systems and space exploration.

As ubiquitous networks of mutually connected devices surround us, it is crucial to understand their security and privacy. The IoT has extensive access to the data and a remarkable ability to influence our lives. A security issue can have a severe impact - not only on privacy or financial losses, but it can also affect human health or even lives. The high amount of cooperating devices makes the security much more complicated. It raises numerous problems to be solved - which participants can we share the data with, which participants may we generally interact with, how to authenticate participants, how to detect a malicious participant, how to introduce a new device into the network, how and when to retire the device, and much more. It is getting further complicated with the environment's heterogeneity - devices in a network have different software versions, operating systems, manufacturers, and often also different owners. The security is typically not a significant concern for all users/stakeholders in the network [1], which means that the security needs to be enforced by the system and must not be left for users to decide. Also, during the early adoption phase, the security is often ignored [2], in order to go to market as soon as possible. Therefore, it is no surprise that security is considered as one of the most crucial challenges [3], [4] of IoT ecosystem.

Generally, security challenges for IoT are similar to traditional applications. However, conventional security architectures were not designed to fully include communication of machines between themselves, typically with limited computational resources, and the heterogeneous and distributed environment of IoT in mind. Therefore standard solutions tend to struggle or even fail, and IoT security solutions must better reflect the specific needs. Another notable contrast from the traditional application of IoT from traditional applications is the altering nature of the environment and fluctuation where devices dynamically connect and disconnect from the network (a churn) and the deployment of the devices and applications in an environment, where we do not have full control of.

One of the properties of IoT environment is its broad access context [5]. The context provides an explanation for the data provided by any participant and allows us to understand the participant's situation better. Moreover, the context could be leveraged to enhance existing security methods with context to provide additional security and to improve or enable personalization. Attempts to leverage context information to enhance “traditional application security" have been here for more than 15 years and are backed up by solid research in this domain [6]–[8]. Thus, my work utilizes the existing knowledge, extends and transfers it to the IoT environment.

This dissertation focuses on authentication, authorization, and partially identity management of IoT devices and users. From the perspective of the standard ISO OSI model, it provides an answer to issues on the highest, application layer. My solution in this thesis aims to provide an easy to use context-aware authentication method(s) for IoT solutions and context-aware authorization architecture tailored for the IoT domain, mitigating current challenges and capitalizing on its advantages.

The specific coals of this thesis are:

1. **Develop method for determining context in IoT environment.** Leverage the extended access to the context and consider specific properties of the IoT devices. The proposed method must be simple to adopt and must be optimized for constrained devices.

2. **Develop context-aware security architecture usable for IoT applications.** Explore existing security architectures and, based on their strengths and weaknesses, propose either a new architecture or an evolution of an existing one. The proposed solution must be scalable and easy to adopt.

3. **Enable security rules sharing across participants in the IoT environment.** Utilize existing tools and protocols and enable quick rule update propagation. Create a mechanism with a single focal point of security administration of the IoT deployment.

The goals mentioned above form a detailed security design that is constructed specifically for the IoT environment. It provides a complete solution from context retrieval, through the security architecture, to security rules synchronization across the network. It allows using only selected parts of the solutions that fit particular needs and replace the other parts with some alternative options. The approach bases on current, existing, and proven solutions. Special emphasis was placed on easy adoption by the system designers, architects, maintainers, and developers. The main advantage of the proposed design is that it will leverage the natural advantage of the IoT environment - access to the context.

Significance: Accomplishing the above-stated goals enables to address security concerns in the IoT solutions. It reduced the work efforts of developers, architects, quality engineers, and system maintainers. Currently, they use either traditional security architectures and approaches or develop their own custom solutions (or, in the worst case, they ignore the security completely). The results of this work provide a complete solution tailored specifically for the IoT environment, leveraging its advantages and mitigating the issues it has.

Scientific merit: The thesis describes a novel and unique method for context retrieval for IoT devices. It is developed with constrained devices in mind and tailored to the computational constraints they have. The data storage can be done on a master device that is controlling the end devices. Further, it defines an extension of traditional security architectures with context-aware elements. The extension is specific, with simple implementation as one of the main characteristics. Therefore, it allows partial adoption of the system's parts where an architect or developer decides and theoretically can be applied with various traditional architectures. Finally, the thesis proposes a method of sharing security rules across the devices in the IoT network.

Broader impact: Results presented in the thesis will contribute to faster adoption of various IoT (often called smart) solutions by allowing developers to focus mainly on the relevant business objectives instead of spending time developing security architecture. It will also help to decrease the number of security incidents. Another benefit of this work is that it will allow developers from other Information Technology (IT) domains to migrate to IoT development easily. Therefore it will further contribute to the spread of the IoT solutions.

Organization of the thesis: The chapter 2 introduces background on IoT. Related work is detailed in chapter 3. The chapter 4 describes the context retrieval method. Context-aware authorization research is presented in the chapter 5. The rule sharing method is elaborated in the chapter 6. Conclusions, contribution summary and the future work opportunities are presented in the chapter 7.

Chapter 2

Background

This chapter goes through the relevant background to help understand the research conducted and put it into the context of the current knowledge. Unlike Related work (chapter 3), this chapter focuses more on common principles, knowledge, and general domain state of the art. The chapter gives a broad overview of the Internet of Things in section 2.1 followed by context-awareness overview in section 2.3 and then it focuses on the general security architectures in section 2.2 and the last discussed topic are context-aware security architectures in section 2.4.

2.1 Internet of Things

Internet [9] origins trace back to the 1980s when computers got connected together for the first time on a bigger scale. Ever since that, new types of devices have been plugged into such networks. It has started step by step with printers and data projectors but since the 2000s, connecting of other devices has ramped up [10]. Today's networks include enormous number of types of "smart objects" [11]. An environment where those devices cooperate together to reach common goals is called IoT [3].

Currently, the number IoT is still expected to grow. It is impossible to get the actual exact number of connected devices, but various industry reports show an increasing trend and predict growth. They vary in the numbers of the devices (as they choose different definitions of them), but the trend is clear. Reports from tech companies illustrate it. Cisco [12] that expects growth from 3.9 billion of devices to 5.3 billion. Gartner [13] that expected 14.5 billion of devices in 2019 to grow to 25 billion in 2021. Intel [14] estimates the number of IoT devices in 2020 to be 200 billion or a very recent business report [15] predicts growth of the IoT market from USD 139.3 billion in 2019 to USD 278.9 billion by 2024, an average yearly growth of 14.9%.

The IoT consists of various elements that utilize connections to the Internet, which share a common goal and cooperate together to provide one or more functionalities. The devices or applications communicate together only through an API, are independent, highly specialized, and frequently owned, created, and maintained by different parties. This is very similar to the microservice paradigm [A.13], [16], which shares a lot of common trains:

1. Microservices (or IoT devices) cooperate together to form complex functionality out of simple features
2. The environment contains heterogeneous microservices deployments (or IoT devices). They have different architectures, programming languages, and paradigms; even different communication channels are possible.
3. There are different creators (or even vendors) for the microservices (or devices)
4. The microservice deployment (or a device) serves a single purpose, it enforces strong encapsulation of functionality, and the communication can be done only through the defined interface

The microservice approach can be used as a starting point for the understanding and evolution of the IoT solutions. The IoT solutions actually use many solutions or technologies that enabled the spread of the microservices couple of years ago. As an example, we can mention the Internet protocol suite, discovery services, or cloud computing. However, some notable differences make the IoT solutions unique. The most notable are:

1. In IoT environment, there is much more devices than services in traditional microservice deployment. The devices are hard, if not impossible, to control. If so, then only in groups rather than every single device.
2. Does not share standard practices (e.g., API's, discovery)
3. Devices can appear or disappear from the network without any backup resulting in unavailable functionality
4. Cost of a single device is small compared to a micro-service
5. Does not enforce clear ownership
6. Lower (or no) control over the IoT deployment environment

The IoT solution deployments are getting increasingly popular. They span across various domains and vary in size, and their production readiness varies from academic or experimental systems through local adoption to large companies or countrywide solution. Here I provide a few examples of IoT applications that are getting tremendous attention:

1. Smart power grid [17]–[21] enables delivery consumption and asset optimization of the grid. It enables to match the demand for the electricity with its supply, and therefore it prevents blackout if the demand was higher and reduces waste, cost, and pollution if the supply was higher.

2. Smart healthcare [22]–[26] puts the main focus on easing the overloaded healthcare systems and therefore saving time, costs, and lives. The predominant approach is home monitoring of patients using smart devices (e.g., wearables). It allows patients to visit the hospital or get specialized treatment at the right time. Alternatively, it can be utilized for rehabilitation, where smart devices can adjust the plan accordingly to the personalized patients' needs and progress.

3. Smart city [27]–[30] includes other IoT applications as smart mobility [31], [32], smart city governance [33]–[35], smart homes [36], [37] or smart power grid. All of this forms together smart city where the technology can either adapt to the flow of the city life or can even optimize it.

2.2 Traditional security solutions

Initially, computers were used as advanced machines to process various calculations or other processes without storing input or output data. While the systems supported multiple users, no data were stored, so security issues were not prevalent. However, when computers began to be used for data management and storage with multiple users accessing the system, the problem of access control emerged.

From the 1970s on, two predominant access control models were used – Mandatory Access Control (MAC) and Discretionary Access Control (DAC) [38]. MAC is predominantly used in applications with strict, centralized access control. Access rules are set by administrators and enforced by the system; users are not allowed to set or modify access policies for system resources. DAC is the opposite; no central element is needed, and each user determines the access policy for resources which they own.

As the complexity of applications increased and evolved into complex information systems with hundreds or thousands of users, a conceptual framework for easier access management was needed. Role-base Access Control (RBAC) [39] allows grouping users together into groups, known as roles; each user may be assigned multiple roles. Access rules are further defined for the roles and not single users. Roles often follow the institution's organizational structure using the information system and are therefore easy to understand for business owners of the application. RBAC was introduced in the early 1990s and quickly became the predominant access control model.

As application user base sizes have continued to grow, the limitations of RBAC have become more apparent, including its unsuitability for context-aware applications [5] or for applications at a scale where the number of roles or role sets needed to cover different access right combinations is too extensive for manual management. Researchers have moved in two directions to address these issues. One direction is to extend the RBAC model in creative and numerous ways [6], [40]–[43]. The other is to develop a more general access control model. Specifically, there is a growing interest in Attribute-base Access Control (ABAC) [44]. It bases access rules on the user's attributes rather than on predefined roles. ABAC can preserve all of the benefits of MAC, DAC, and RBAC while adding more flexibility – it can be used to support, or be implemented under, any of these access control paradigms.

The access control methods described above deal predominantly with authorizing users to access specific resources or take specific actions, rather than describing how the user should be authenticated; authentication is considered a prerequisite for authorization. This authentication may be accomplished using three basic credential categories. The first category, "Something I am", represents properties about the user, including their location or biometric characteristics. "Something I have" stands for credentials that were given to a user; the user possesses the credential. This category includes all types of keys, tokens, cards, or even personal devices like phones. The last and most familiar category is "Something I know", most often represented by passwords, but not limited to them – it also includes the user's knowledge of security questions, their interaction history, and other information.

Authentication credential categories may be combined together for increased security or to improve the user experience. Multi-factor authentication is a common practice to increase security and prevent a breach in the event that a single credential is compromised. Some authentication frameworks provide single sign-on functionality where users sign into a trusted authentication provider using their credentials (often just a password) and receive provisional tokens, which are then used to authenticate against other services. Subsequently, those services verify the token with the authentication provider and log the user in. Often, the usage of the token is automated, and the user only needs to log-in once.

Identity management is closely related to authentication and authorization. A virtual identity must exist against which users may authenticate and which stores user attributes (unique identification, attributes as understood in ABAC, RBAC roles, and other required information) used for authorization.

At the most basic level, applications each manage identity independently, using as little information as possible – generally, this includes both a principal (identity unique identifier) and credentials used for authentication. As applications become more complex, the information required for user authorization grew to include roles or identity attributes. As the number of applications per user and the number of users per service increase, it becomes difficult both for the user and service administrators to manage the growing amount of identity information required. These developments led to the need for federated identity management – a way of providing identity services for multiple applications, often tied to authentication mechanisms. Currently, several implementations of federated identity management exist, including using LDAP [45] for identity management or using OpenID [46] as an identity service.

2.3 Context-awareness

A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task. This is the definition of context-awareness by Dey [47]. To understand it, we need to explain what the context actually is. Context definition by Abowd [5] is the most prevalent and cited, and he defines context as: *context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.*

Abowd published his context definition in 1999. It was the time when the first context-aware applications were developed. For example, Chen and Kotz in the article "A Survey of Context-Aware Mobile Computing Research" [48] from 2000 presents more than ten context-aware applications and summarize multiple methods for sensing the context, modeling the context and architecture proposals. Harter et al. in 2001 propose in their architecture [49] to use data from external sensors. A massive increase of usage of the context-awareness occurred when Web 2.0[50] took popularity and mobile applications started to gain attention [51]. With Web 2.0, users started to participate in the website's content, which led to the possibility to retrieve more information about them. Using mobile applications leads both to the ability to extract extra contextual information, like exact position, and the users' demand for personalized applications.

The process to represent the context in the application typically consists of two steps - model the context, define its types, attributes, relationships, and data quality characteristics. The existing data can then be categorized, and mainly all the newly acquired data will comply with the model. The most frequent modeling techniques are explained in [48] and [52] and include object key-value, markup schemes, graphical, object-based, logic-based, and ontology-based modeling.

Currently, context is used in many application and fields as transportation [31], [32], [53] and all "smart" solutions which include, smart health[22]–[26], smart homes [36], [37], smart power grid [17]–[21] or smart cities [27]–[30]. Users do not take context-aware application as something extra, but they require it as a standard. People want to get personalized results of the search. They want to get their route planned with considering traffic situation [53]. They want their sport tracker to adjust their training plan based on their health condition, weather, and previous workout results.

We can categorize usage of the context into the three categories [47]:

1. Presentation of information and services to a user - system uses context to provide more accurate information to a user. An example of it is a personalized search on search engines.

2. Tagging of context to information to support later retrieval - the system automatically adds some additional contextual information to the data that are added by the user. For example, the location of the user and the time when the user data.

3. Automatic execution of service for a user - system determines when it is appropriate to launch some service based on a context. It can be a recalculation of a route when the user leaves the path, triggering of light when he enters a room, or in computer security launching additional authentication based on the user's location history.

One of the main challenges of context-aware applications is context retrieval [54].

Some information is evident for the system (e.g., time, frequency of log-ins, history of application to user communication), others can be guessed but not guaranteed (e.g., geographical location determined from the IP address). In contrast, much information is difficult to obtain (e.g., biometric information about the user). All the information about the user's context may significantly increase the security of the system, and at the same time improve the application's user experience.

2.4 Context-aware security architectures

Security architectures can also benefit from the context-awareness. Context-aware elements bring benefits for both the user and owners/maintainers of the application. The application user is presented with a better user experience, and the application owners achieve higher security of their system if context information is considered for the security.

Traditionally, security is rather a conservative part of the systems or applications. It can be either because there is a need for a proven and well-working solution or because it is not facing the users, and there is not intense pressure for modern fancy solutions to be presented as a differentiating feature. Also, the users do not expect the security to contain some novel approaches. Therefore, the context-aware security architectures' adoption (and readiness generally) is behind the general context-aware solutions.

Usually, users are assigned various roles in application or permissions for resources, and security rules are independent of context. We can expect that users and application owners would benefit from application security based on context to provide specific access to resources based on context. Applications using context-aware security can be much less obtrusive for users. They can be asked for different authentication methods based on context; they can be authorized for the same resource in various ways depending on their context. For example, access from City A can have different access rights than access from City B. They can even sometimes omit authentication because their context is trustworthy by itself (e.g., access from the company workplace). Similar to users, also application operators can profit from context-based authentication and authorization. They might define more strict security rules for suspicious users' behavior (e.g., Internet access to system confidential resources at night). Using a context allows system administrators for more fine-grained security rules, which would be otherwise tangled through multiple rules and make them unsustainable for maintenance.

When adopting context-aware security architectures, two basic approaches are possible. Either extend and adapt some existing security architecture for context-awareness or develop entirely new architecture. Solutions from both categories were explored and described in the literature, though the adaption of some proven architecture is more common.

Out of the traditional architectures, it is easiest to adapt ABAC. It can either work out of the box, making some of its attributes contextual. E.g., location, time, temperature, and other attributes can be easily ported in the ABAC attribute system. For more complex contextual information is possible to extend it [55].

Extending RBAC requires more effort and is not that straightforward. Therefore there are various paths on how to achieve context-aware RBAC. One of the approaches is to add another set of roles to RBAC. Moyer et al. [56] propose creating two additional sets of object roles and environmental roles and tying permissions with a trio of roles. Further research [57] simplify that to just one additional set of environmental roles. They are hierarchically composed and represent the current state of the system. Similarly to this approach, it is possible to have an additional set of context roles [6]. Slightly different method is to introduce concept of trust and extend the simple RBAC with it[7]

A different method is to grant roles after user during authentication based on his context [43]. That way user can obtain new roles, which are reflecting his context. The idea is further developed by into Context-Aware RBAC [58]. It also allows roles to be granted based on context, but there is a second layer of authorization architecture, which is responsible for granting and revoking roles when the context changes, and therefore roles are dynamically reflecting context.

There also exists a possibility to solve that problem by adding another element not based on roles. An example is adding context constraints to security policies [59]. When the permission is checked, a user needs to possess not only the permission for the resource (based on his role) but also fulfill context constraints. Similar approaches are to introduce other system participants into the system. Either they determine the access rights on those four elements: permission, role, context, and authentication method [60] or alternatively, it can use four different context actor - context owner, context provider, context broken, and context-aware service [8].

There are significantly fewer security architectures that are not strictly based on a pre-existing solution. One of the methods that might work with every security architecture is to add additional context dimension to current security rules [61]. Another remarkable idea is to assign permissions to directly contexts [62].

2.5 Internet of Things and Context-awareness

IoT consists of various participants. They range from real human users through applications that aggregate the information and take actions based on them, to relatively small end devices that interact with the real world. Those devices often contain various sensors that capture their surroundings from the real world - the context. That contextual information is then used to present the information to the user, the execution of other services, or control IoT devices, or tagging the data for later auditing, statistical or other use.

Without the use of contextual information, the whole idea of IoT does not make sense. The solutions are based on some kind of smart behavior, driven by both user inputs and environmental changes. Smart power grids could not work if they did not have contextual information about the power grid and possibly about the weather or current date and its meaning (e.g., school holidays or public events that will change the consumption schema). Smart transportation solution with no access to traffic information and sources and destinations and preferably real-time location of the participants is also meaningless.

The idea of context-aware systems having access to information from sensors is not novel; actually, it is more than a decade old [54]. The initial solutions that we could consider to be predecessors of IoT solutions are sensors networks [63]. The difference between IoT and sensors networks is very blur, but generally, the IoT contains sensor network and other aspects that make implementation more manageable, cheaper, often more extensive, and generally more feasible.

Everything that applies to traditional context-aware applications also applies to IoT solutions. The main difference is the excellent access to the contextual data, architecture that is based primarily on the contextual data, and that relies on it and the capability to deal with ambiguity and heterogeneous environment.

In the IoT solutions, there is no standardized or “best” approach for almost none of its challenges - how to model context, what principle/architecture is best to obtain it, what is the best reasoning model. The acquisition of context can range from direct access to sensors through using various middleware solutions aggregation contextual information from a specific part to a big contextual data lake [64]. The reasoning models [65] [66] can be rules-based, build on top of supervised or unsupervised learning, ontology-driven or probabilistic reasoning.

A great resource to get more insight is the survey article [67] describing specifically the IoT context-aware computing providing an in-depth overview of this topic. It includes network architectures, open challenges, context types and categorization, levels of context-awareness for various systems, context management principles, context acquisition techniques and their lifecycle, context models, and already existing contextual systems.

Chapter 3

Literature review

This literature review's motivation is to provide an overview of current research progress in the domain of IoT security. This is a broad discipline, and therefore I focus mainly on authorization, authentication, and identity management papers, specifically at the highest layer of the network stack, typically the application layer. While "network stack" is not the precise model used for the IoT, I use the term in lieu of a more standard vocabulary to describe the IoT technology and communication architecture; there does not yet appear to be joint agreement on such a term. I am interested in architectures, projects, solutions, proposals, identity-management of IoT devices and frameworks dealing with user-to-machine and machine-to-machine authentication and authorization as those topics largely overlap with my dissertation research. Candidate papers are identified not only by a manual survey but also by a systematic search [68] through major indexing sites and portals. The resulting papers are analyzed to provide a comprehensive overview and classification of existing work.

The chapter achieves:

- Categorize the security solutions and provides their taxonomy
- Identifies context-aware security solutions and goes through their methods
- Examines whether the IoT solutions are already existing solutions adapted for IoT environment or novel methods are proposes.
- Explore the security solution's architecture in terms of whether the security solution is centralized or distributed.
- Enumerate the existing solutions to find out whether they are focused rather on User to Machine (U2M) or Machine to Machine (M2M) interactions

I have conducted the literature overview in late 2017, and it led to an article [A.2] that was published in early 2018. The article had considerable impact on the scientific community, as it obtained 27 citations, 9 of them from journals indexed in Web of Science (WoS) Science Citation Index Expanded (SCIE) (with impact factor). The chapter is based on the article, but it is greatly enhanced with the latest research conducted and published in the last three years.

If the reader wants to get more familiar with the whole broad topic of IoT security or wants to read additional materials providing an overview of the research problem, I list some great surveys or systematic study papers from recent years. Noor et al. published broad IoT security survey [69], and I consider this study to be excellent, though limited only to years 2016 – 2018. The most recent overview is provided in [70], published in July 2020. Milovlaskaya et al. summarized information security research in [71]. Survey of the continuous authentication methods [72] provides a great overview of the specialized issue. Another focused study [73] goes through industrial IoT security issues.

3.1 Search

This chapter primarily uses data from the survey article [A.2] that contained data from 2017 and earlier. However, during the time before writing this thesis and publishing the article, there have been a significant amount of newly published papers, and therefore I update the initial set of papers with the newest scientific results.

3.1.1 Initial Search

In order to systematically review all existing research and answer our research questions, I performed searches at the following indexing sites and portals: IEEE Xplore, ACM Digital Library (ACM DL), WoS (Core), SpringerLink, and ScienceDirect.

To show that my search queries provide results relevant for this dissertation, I evaluated the search query results against a control set of papers identified as matching the scope through manual search before I performed the search queries. When a search query returned papers from the control set, this is evidence of the search query's usefulness.

The search query consists of two parts. The first part targets terms and keywords to be included in the paper, and the second part removes papers that contain terms we are not interested in. Naturally, I am interested in research about the IoT, so I

Indexer	Query
General query	("Internet of Things" OR "IoT") AND "Security" AND ("Authentication" OR "Authorization" OR "Identity" OR "Access control") AND NOT ("Network" OR "Hardware" OR "RFID" OR "Protocol" OR "Cryptography" OR "Survey" OR "Study")
IEEE Xplore	("Abstract": "Internet of Things" OR "Abstract": "IoT") AND ("Abstract": "Authentication" OR "Abstract": "Authorization" OR "Abstract": "Identity" OR "Abstract": "Access Control") AND "Index Terms": "Security" AND NOT("Index Terms": "Network" OR "Abstract": "Hardware" OR "Abstract": "Cryptography" OR "Abstract": "Protocol" OR "Document Title": "Survey" OR "Abstract": "RFID" OR "Document Title": "Study")
ACM DL	Abstract:(IoT "Internet of Things") AND Abstract:(Authentication Authorization Identity "Access Control") AND Title:(-study -Survey) AND Abstract:(-Hardware -rfid -Cryptography) AND Keyword:(-Hardware -Physical -Network)
WoS	TI=(Internet of Things OR IoT) AND TS=(Authentication OR Authorization OR Identity OR Access Control) NOT TS=(Hardware OR Cryptography OR Protocol OR RFID OR Physical OR Network) NOT TS=(Survey OR Study) AND TS=Security
SpringerLink	(Authentication OR Authorization OR Identity OR "Access Control") + title ("Internet of Things" OR IoT)
ScienceDirect	TITLE-ABSTR-KEY("Internet of Things" OR "IoT") AND TITLE-ABSTR-KEY(Authentication OR Authorization OR Identity OR "Access Control") AND KEY(Security) AND NOT (TITLE-ABSTR-KEY(Hardware OR Cryptography OR Protocol OR RFID) OR title(study OR survey) OR key(Physical OR Network))

Table 3.1: Queries used for the search

include “Internet of Things” or “IoT” as one of the main groups. Another important term is “Security” as I target only those papers that deal with security. Further restriction terms refine the results to include only papers with “Authentication”, “Authorization”, “Access Control” or identity management, which is shortened to “Identity”. The second portion of the query is to limit the number of articles in the result set. I removed papers that deal with the security at the lower levels of the network stack. This translates to the terms “Network”, “Hardware”, “RFID”, and “Protocol”. Cryptography is not a particular focus of this survey, so I also remove research with this keyword. Finally, I remove papers that are surveys themselves, containing “Survey” or “Study” in their title.

The query syntax differs for each indexing site, but I aimed to search through abstracts or keywords/topics where applicable. The queries are constructed as similarly as possible. The exact queries used, including the general query I used as

Indexer	Results	Prefiltered	Relevant
IEEE Xplore	120	29	14
ACM DL	84	9	7
WoS	67	31	13
SpringerLink	33	8	6
ScienceDirect	27	9	2
Total	331	86	42

Table 3.2: Number of articles processed in the survey

a template, are listed in Table 3.1.

I encountered an issue with the search function in SpringerLink. The search system is not able to process an advanced query, such as the one I designed. I used a more straightforward query that returned 383 papers and processed these results by constructing a short script that opens the particular page for every exported paper, extracts the abstract, and performs the advanced query locally on our machine.

Running the query across all five indexing services gives us a set of 387 papers, from which I exclude those with less than four pages. Since WoS indexes papers that appear at other sites, it contains 16 duplicate papers, which I also remove. As a final filter, I read each article's abstract and removed those papers not within the designed scope; this gives me 86 prefiltered candidate papers. I also exclude [A.5] as it is discussed in a separate chapter.

These remaining papers I read one by one, with some exceptions. The full-text of one paper could not be downloaded; this was removed from the results set. Three of the papers were highly-similar extensions of another paper in the results set. In this case, I used the extended paper and discarded the shorter versions. I also removed papers that did not fit into the literature review's scope – those where the abstract initially indicated a connection to our research questions, but the full text did not. The complete statistics of papers found, prefiltered, and included for every indexing site can be seen in Table 3.2.

3.1.2 Update search

The initial idea was to update the research with the same approach, just for the years 2018 – 2020. However, this turned out to be unrealistic; the amount of the research in the area of IoT security has multiplied. There are currently five times more research publications than three years ago. Table 3.3 illustrates the growth of the research. I have intentionally skipped SpringerLink as it requires

Indexer	2017	2020	Growth
IEEE Xplore	120	507	387
ACM DL	84	511	427
WoS	67	349	282
WoS SCIE	21	155	134
ScienceDirect	27	171*	144*
Total	298	1537	1241

Table 3.3: Growth of the publications

Primary source	count
IEEE Xplore	7
ACM DL	0
WoS	8
Springer	4
ScienceDirect	2

Table 3.4: Primary sources of publications

post-processing on the computer. ScienceDirect has changed the search to allow a maximum of 8 Boolean operators, and therefore its results contain a larger set of articles and * marks the numbers. Last note is that I included both WoS Core collection and separately SCIE index. In the Total row, I use only the larger WoS Core collection as it is a superset of the SCIE.

I have decided to go through only WoS SCIE articles to extend the initial set. The reason is the vast majority of useful articles are indexed in the WoS SCIE (journals with IF). Also, those articles typically have the highest impact on the scientific community (measured by citations).

The statistic is the following - out of 155 articles, I have filtered out 67 based on abstracts that I have read. I have found 21 articles to be related to the dissertation topic. This is significant growth, as in the initial search, there were only 11 articles indexed in WoS SCIE. Table 3.4 shows the distribution of the primary sources and suggests that 13 of the found articles would duplicate other indexing services. One additional note is that 18 additional papers were variations and adaptations of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) for IoT that is not precisely the dissertation topic, but it is somehow related as it can be understood as a mean of authorization.

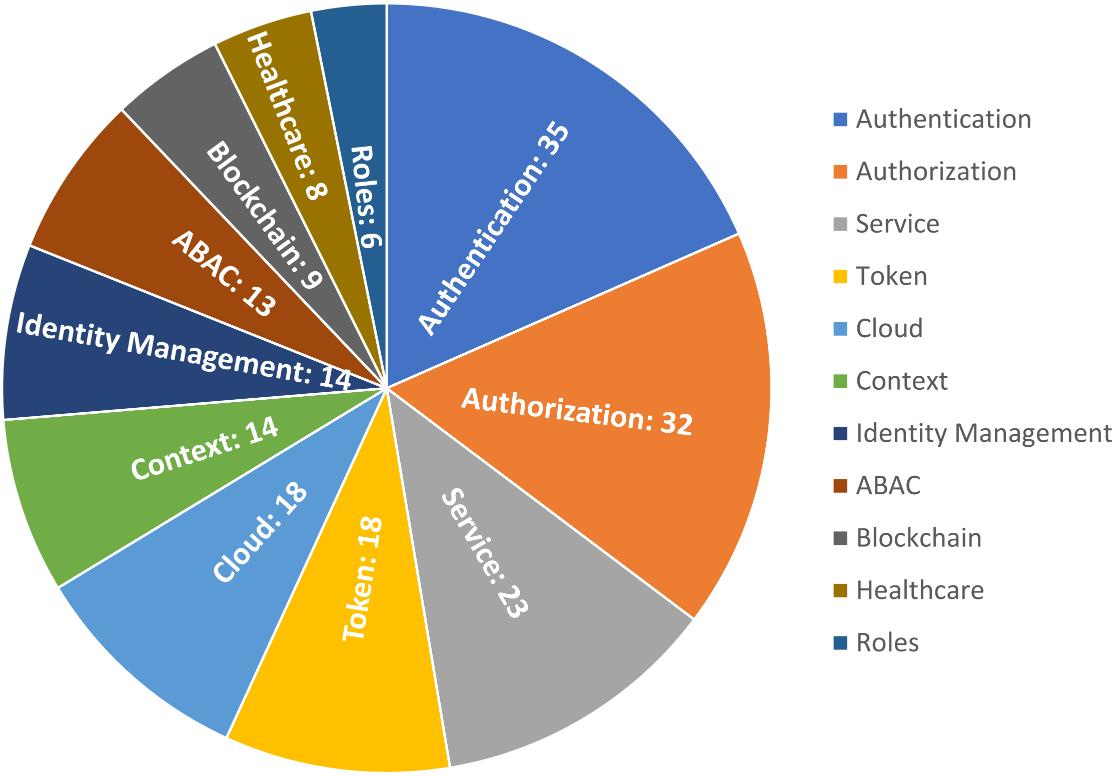

Figure 3.1: Number of keywords found across all articles

3.1.3 Final result set

In the final set that is used for the statistic and overview further are both searches combined. Together it is 63 related publications that are categorized and described in the following sections. If it is required for better comparisons, I only use the 11 articles from WoS SCIE in the first result set.

3.2 Taxonomy

To find candidate categories based on the most prevalent keywords, I employ the RAKE [74] algorithm for keyword extraction. First, I transform the PDF documents using pdftotxt [75] and strip references or appendices. Then, I apply the RAKE algorithm with the following parameters for the keyword extraction: at least five characters, a maximum of two words for the keyword, and at least four occurrences in the text. For each keyword, I then find matching articles. Only keywords present in at least two papers are taken into consideration. I then group synonymous keywords into categories. As a consequence of this approach, a paper may fall into multiple categories.

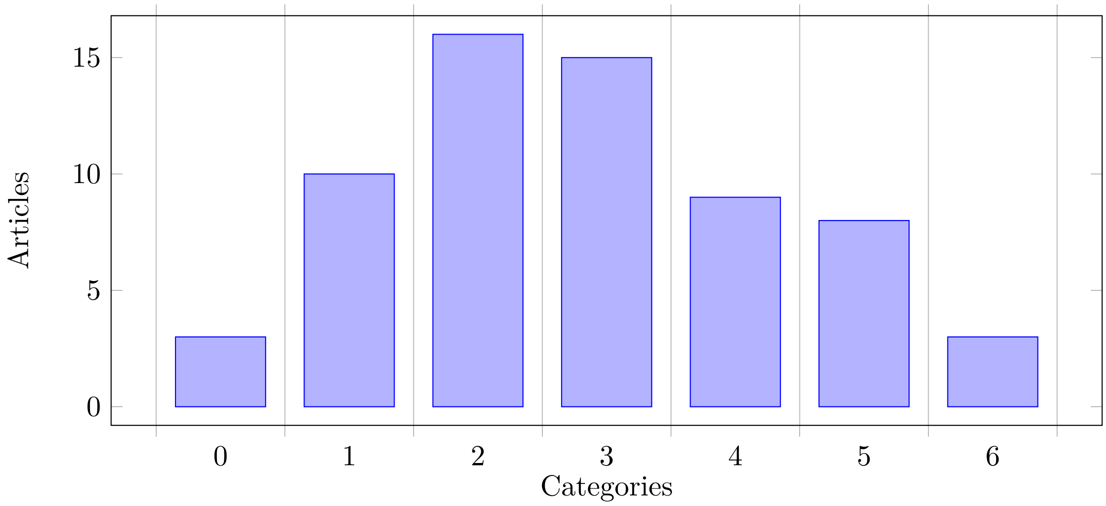

Figure 3.2: Number of categories suggested by RAKE per article

The results (excluding general terms) suggest the following categories of the papers. They are also illustrated in Figure 3.1

- **authentication:** papers that addresses authentication [76]–[110]
- **authorization:** articles dealing with authorization [80], [82], [83], [85], [88], [90], [91], [93]–[95], [97], [100]–[102], [105], [107], [109]–[124]
- **service:** solutions that can be used both in IoT and Service Oriented Architecture (SOA) [77]–[79], [83]–[85], [87], [89], [91], [93], [96], [98], [100], [102], [106], [108], [111], [121], [125]–[129]
- **token:** articles that use any form of token as an information bearer in their proposal [81], [83], [85], [96], [100], [103], [106], [109]–[112], [115], [116], [118], [121], [122], [125], [130]
- **cloud:** research addressing security issues of cloud-based IoT devices [76], [78], [82], [97], [98], [104], [107], [109], [116], [120], [129], [131]–[133]
- **context:** papers using or proposing context-aware methods [76], [85], [94], [96], [102], [103], [110], [111], [113], [124], [125], [131], [132], [134]
- **identity management:** solutions discussing identity management [77], [80], [84], [95], [96], [99], [101]–[103], [107], [108], [110], [122], [125], [126], [129], [130], [135]
- **attribute-based** subset of authorization proposals that involve ABAC [77], [90], [93], [95], [100], [103], [110], [113], [121], [122], [124], [132], [135]

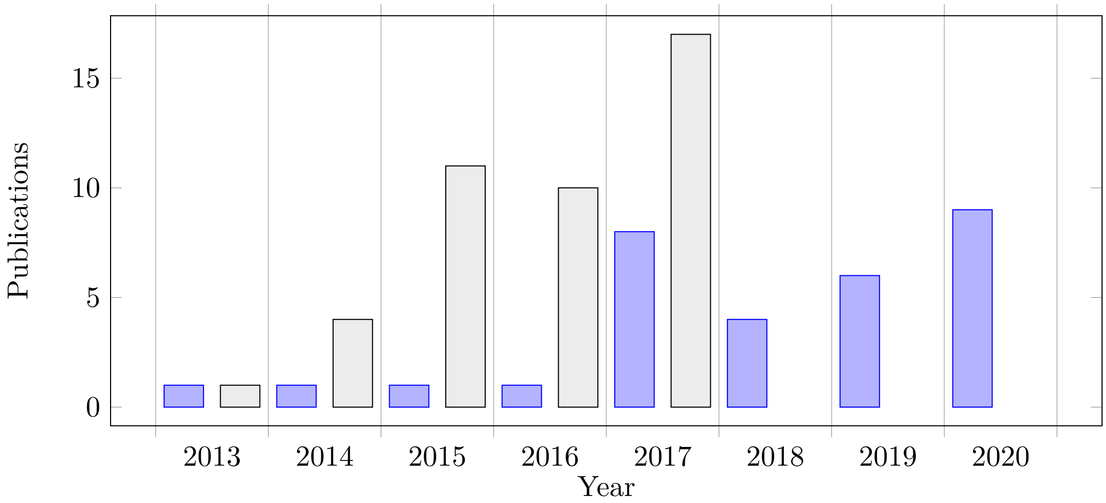

Figure 3.3: Number of publications per year

- **blockchain:** research that utilized blockchain [94], [105], [107]–[109], [122]–[124]
- **health care:** projects that specifically address the health care domain [80], [84], [90], [93], [105], [116], [124], [132]
- **roles:** subset of authorization proposals that involve RBAC [80], [90], [100], [123], [124], [136]

Two of the papers do not fit into any of the above categories [137], [138]. One article [137] is likely too short for RAKE to perform any meaningful analysis; I have not identified any apparent reason why [138] is not categorized by the algorithm. Nevertheless, both of the papers address authentication, and I have included them in this category.

In total, slightly over 50% of the articles gets two or three keywords. A significant number of research papers fit into one of four categories. Two papers did not fit any category, and another three fit to five categories. This statistic is shown at Figure 3.2. As illustrated in Figure 3.3, the research covered by this survey shows an evident increase of interest in IoT security based on the number of articles published. The grey bars represent the initial data set; the blue bars are articles indexed only in WoS SCIE (manually extracted from the first set and combined with the update search). The chart illustrates steady growth, except for 2017, which had a significant amount of publications.

The authentication, authorization and services categories are described in the subsequent subsections as they are the most populous categories. Identity management is also described in a separate section, as it is closely related to the authentication as well as to my research. Articles with context awareness elements are further described in their own section.

3.2.1 Authentication

Authentication is addressed by 36 papers from our pool – more than half of the articles in the survey. Authentication is the process of confirming an attribute claimed by an entity. In the vast majority of cases, it is confirmation of an identity that the entity claims using credentials.

Traditional authentication methods, enhanced with multifactor authentication based on a location, are described in [76]. Their system considers user location, and they develop an additional factor for multifactor authentication, which ascertains the physical possibility of a user being in a particular location, e.g., a user cannot possibly be in Los Angeles if they just logged in from New York. This adds additional security without requiring the user to perform different actions.

In [77], the authors suggest enhancing privacy during authentication by basing authentication on attributes rather than identities. A trusted authority issues certificates which prove that an entity possesses a particular attribute; these certificates are used for authentication when communicating with other services. This scheme preserves both entity privacy and the advantages of centralized identity management.

The authentication model for cloud-based IoT is elaborated by Barreto et al. [78]. Their solution supports two authentication stages: one for basic and a second one for advanced access, e.g., administrative purposes. They do not describe specifically how the authentication should be done; instead, they specify methods that cloud services should provide for authentication.

To achieve efficient and smart authentication of IoT devices, Cagnazzo et al. [79] suggest using Quick Response (QR) codes; specifically, XignQR [139]. Every device has a printed QR code that contains important information about it, e.g., an ID representing its service provider, authentication server address, and digital signature. Scanning the QR code and sending it to the authentication manager allows the manager to decide which authentication method it should enforce on the user. This approach can be useful when physically managing large amounts of devices simultaneously, e.g., in a medical environment or a factory.

A security framework following the Architecture Reference Model (ARM) [140] is described in [83]. It bases authentication on the Extensible Authentication Protocol (EAP) over LAN [141]. EAP is widely used and recognized as a mechanism to provide flexible authentication through different EAP methods. Those methods allow an EAP peer to be authenticated by an EAP server through EAP authentication for network access. While their work proposes interesting solutions, they do not provide any case study or usability study.

Kumar et al. [84] assume that the best authentication method for wearables and nearables (devices that are not worn but are generally close to the user) are the biometric information of their owner. The proposed solution requires the user to register their biometric characteristic(s) in person with the authentication provider. Later, access points close to the user – wearables or nearables – capture the user's biometric information and authenticate them by comparing those characteristics with the registered characteristics. However, there is an issue with privacy as many users are reluctant to share their personal information. A slightly different method is to measure the user's gait and authenticate the user based on it [99], [104]. The initial gait is trained on the 1-minute walk. The method's innovation is that it improved accuracy by speed adaptive methods and smart threshold calculation for gain template matching. Another very different method of authenticating using biometric information is presented in [98]. It proposes to use brainwaves for authentication. They show users various images that they are either familiar or unfamiliar with and measure their reactions through brainwaves.

Three almost identical works proposed the OpenID protocol as the authentication method in the IoT environment [85], [100], [106]. They describe a central service issuing tokens and communicating through a RESTful API [142] over the HTTP(s) [143] protocol, allowing rapid development and acceptance among IoT devices as all technologies used are proven, well-documented, and widely supported. A downside is that the OpenID protocol was not designed with IoT usage in mind and can be more demanding of computation and network resources than specialized protocols.

Another framework [86] for authentication is formally described using process algebra, specifically CSP [144]. The framework contains three authentication forms. An *entity* authentication is the capability of verifying the identity that the entity claims. An *action* authentication refers to the authentication of the actions of devices and whether they are allowed. A *claim* authentication verifies the authenticity of devices' claims about previous actions. It also has three strength levels for each form: weak, non-injective, and injective. The paper does not provide any proof of concept or another kind of demonstration of their solution.

A mechanism of HTTP(s)-based authentication for IoT devices using a hash-chain generated between server and the client is explained in [87]. This hash-chain is generated during the login process and serves as a One Time Password for the client to authenticate against services. If a device does not have the required capabilities (e.g., battery lifetime, computational power, a network connection) to generate the hash-chain, or those capabilities are in use for other functions, another device acting as a proxy may be used to generate the hash-chain.

Continuous authentication of personal IoT devices is addressed by Shazad et al. [88]. Current practice is to authenticate an entity just once when a session is established and keep them authenticated until some timeout occurs or the session is otherwise closed. This session persistence presents a potential security risk. The authors divide devices into two categories – those which maintain physical contact with the user and those which do not. Devices that keep contact can be authenticated using various biometric information, both direct (blood flow rhythm) and indirect (using inertia measurement unit to check a user's gait). The authors propose using radio frequency signals for devices that are not in physical contact with the user. For example, Wi-Fi signals are reflected by the human body, and the resulting distortions can be measured and used to determine users' walking speed, gait cycle, and other physical properties. Different for continuous authentication of users is presented in [102]. It describes users' context-aware authentication (and authorization) based on their behavioral patterns observed through IoT devices. The confidence manager does the authentication, and then the results are used both in the authentication and authorization process.

Advanced authentication methods better than the current approaches are suggested in [89]. Most of the traditional methods have flaws or were not designed to be frequently used (e.g., passwords – almost no one can memorize strong and unique passwords for every service or device they use, so users reuse their passwords). Their proposal is based on users' digitized memories. Users would authenticate themselves against their digitized memories based on date and time, place, people or pets, devices, habits, audio, or ownership recognition. They map different suitable methods, including choice selection, alphanumeric input, image part selection, or interactive categorization.

Article	Centralized	Decentralized	U2M	M2M	Context-aware	Specifics
[76]	Yes	Yes	Yes	No	Yes	Service answering whether user can be in the given location
[77]	Yes	Yes	Yes	Yes	No	Use of attributes for authentication
[78]	Yes	No	Yes	Yes	No	Authentication through cloud
[79]	Yes	Yes	Yes	No	No	Reading QR codes physically present on a device
[80]	Yes	Yes	N/A	N/A	No	Framework designed to preserve patient privacy
[81]	Yes	No	Yes	Yes	No	Adjustment of Web API management; OpenID Connect
[82]	No	Yes	Yes	Yes	No	Authentication for devices with constrained computational power
[83]	Yes	No	No	Yes	No	ARM compliant; EAPoL; RADIAL
[84]	No	Yes	Yes	No	No	Biometric from wearable and nearables
[85]	Yes	No	Yes	Yes	No	OpenID Connect
[86]	Yes	No	Yes	Yes	No	Authentication framework mathematical description using CSP algebra
[87]	Yes	No	Yes	Yes	No	HTTPS-based device authentication using hash chain as One Time Password
[88]	N/A	N/A	Yes	No	Yes	Biometric; continuous authentication
[89]	Yes	No	Yes	No	Yes	User's electronical history
[90]	Yes	No	Yes	Yes	No	Authentication based on attributes
[91]	N/A	N/A	No	Yes	No	WS-Security adaptation for IoT
[92]	N/A	N/A	Yes	No	No	One time passwords using words chosen by a user
[93]	Yes	No	Yes	Yes	No	Full security framework
[94]	No	Yes	Yes	Yes	No	Blockchain access control framework
[95]	N/A	N/A	No	Yes	No	Authentication on perception level
[96]	No	Yes	Yes	Yes	Yes	Privacy preserving based on partial identities
[97]	Yes	No	Yes	No	No	Smart home, FIDO
[98]	Yes	No	Yes	No	No	Privacy preserving based on partial identities
[99]	Yes	No	Yes	No	Yes	Authentication based on gait
[100]	Yes	No	Yes	Yes	No	OpenID Connect
[101]	Yes	No	Yes	Yes	No	Authenticaiton based on functional right
[102]	No	Yes	Yes	Yes	Yes	Continuous context-aware authentication
[103]	Yes	No	Yes	No	Yes	Confidence score calculated from context
[104]	Yes	No	Yes	No	Yes	Gait analysis; ECG like signal processing
[105]	No	Yes	Yes	Yes	No	Blockchain
[106]	No	Yes	Yes	Yes	No	OpenID Connect
[107]	No	Yes	Yes	Yes	No	Blockchain
[108]	No	Yes	Yes	Yes	No	Blockchain
[109]	No	Yes	Yes	Yes	No	Blockchain
[110]	Yes	No	Yes	Yes	Yes	XACML

Table 3.5: Summary of authentication articles

Wiseman et al. [92] present a niche but interesting problem along with a solution. They address the issue of pairing an IoT device with its "master" account. Connecting from devices using a password can be difficult or even impossible because of the lack of a proper input method. One method to avoid this is to let the device display an access code and add the access code to the master account. They examine this process from a user experience perspective and compare convenience between alphanumeric codes and codes generated from human-readable words.

A privacy-preserving, decentralized identity management framework for the IoT is presented in [96]. Identity in the IoT is extended not only to users but also to IoT devices themselves using an ARM-compliant, claims-based approach built on top of Identity Mixer technology [145]. They define partial identities as subsets of a user or device's virtual identities that preserve privacy while being sufficient to provide identity confirmation. They show the use of their framework with Distributed Capability-Based Access Control [83]. Identity attributes are disclosed by specific proof and are employed during authorization based on XACML rules to obtain capability tokens used to access a service.

Khalid et al. [107] decentralizes authentication using blockchain technology. There is a fog layer for every domain/application to allow authentication (and possible authorization rules storing) of the devices. When the device connects to a network, it finds a close fog authentication server and authenticates through it. It receives a private key, and a public key is stored in the blockchain. Devices can communicate only with devices that are authenticated, and their identity is propagated in the blockchain. Similar, blockchain-based, approach is used in [108]. It uses multiple blockchains for communication of IoT devices, where there are multiple local blockchains and a single global one. It categorizes devices into simple devices, proxy nodes, and manager nodes. Proxy nodes authenticate (and authorize) near constrained devices and use local blockchain for it. The local blockchain is restricted to a specific application or deployment. If a device wants to communicate with a device outside of its network, it uses a manager node that is part of the global blockchain. Another similar blockchain method is proposed by Pallavi et al. [109], which also uses a fog layer.

Finally, there is a group of papers [80]–[82], [90], [91], [93]–[95], [97], [101], [103], [105], [110] that address authentication tangentially either as part of an broader and more complex framework or project, or to solve authentication issues as a side effect of dealing with another problem.

Table 3.5 presents an overview of authentication research, reflecting the information I extracted from the papers. It shows which solutions support centralized and decentralized architectures, which are U2M or forM2M communication and which posses at least some elements of context-awareness.

3.2.2 Authorization

Authorization is the process of granting permissions to execute specific actions to given entities – in our scenario, specifically to users, devices, or applications. There are a total of 32 articles in the identified pool addressing this topic. Authorization category ties with services as the second most populous category.

Access control based on trust in an ARM-compliant model is proposed by [111]. It describes various levels of trust, a multidimensional attribute that describes various concerns in the network. The authors specify dimensions: quality of service (including network availability and throughput), security (e.g., authentication and authorization protocols, encryption.), reputation (recommendations from other devices), and social relationship (the group or groups of IoT devices to which an individual device belongs, e.g., those made by a specific manufacturer or currently in a particular location). This trust is used for final authorization within the environment.

The authors of [80] describe a complex framework for use in the healthcare field. They employ a version of RBAC where a user, specifically a patient, grants permission to access his data based on a particular role – a group of doctors and nurses. A centralized authentication server enforces the resulting security rules.

Another paper [112] develops an authorization architecture based on IoT-OAS [146], authenticating users using tokens similar to those used in OpenID. Every device has a designated owner and a set of actions or permissions. Users may request and share permissions with one another; multiple operational cases are described in the paper.

Gerdes et al. [82] tackle the problem of authorization and authentication for devices with constrained computational power. The authors divide IoT devices into the categories "constrained" and "less-constrained" based on resource availability and allow less-constrained devices to perform some authorization functions on behalf of the constrained devices. The paper includes basic methods for these authorization management tasks, and "principal actors", which represent the person or company that owns the specific device or the data on the device, must set appropriate policies for each situation about which tasks can or cannot be offloaded.

One solution to the problem of data access control across a shared network is developed in [113]. The authors use Ciphertext-Policy Attribute-based Encryption

[147] and enhance it with a set of policy descriptions in a eXtensible Markup Language (XML) file. Access policies are based on entity attributes and structured as a binary tree with “And” and “Or” operations available. Entities present a keyserver with a list of their attributes, and the keyserver generates a key that can only decrypt data to which the listed attributes allow access. A similar approach is discussed in [101] that is aimed to reduce privileged access. It suggests to give access to functionalities rather than assign roles/attributes. Functionality consists of two elements - data type and allowed actions for them. The rules are enforced by identity-based encryption [148] performed on a cloud server.

A framework introduced in [83] supports not only authentication but also authorization, enabled by creating an Authorization Server which issues access tokens according to security rules stored in XACML [149], an XML schema for representing authorization and entitlement policies. Entities request authorization tokens based on their attributes and then use the tokens to access services provided by or data stored on another server or device.

Kurniawan et al. find classic security strategies unsuitable because they are centralized and scale poorly in the IoT environment. They propose a trust-based model [114] based on Bayesian decision theory. The authors compute Bayesian trust values based on three inputs: experience (the history of interactions between the actors), knowledge (what is already known about the entity and the context), and recommendation (how much trusted peers trust the entity in question), and use these trust values as input to a loss function that determines the cost of an action. Access control decisions are made based on the loss function’s output, given a particular trust value.

Numerous proposals based on the existing OAuth protocol [150] use tokens that encode the access rights (e.g., roles or attributes) of the token owner and a configurable lifespan. Some methods [100], [116] use JSON Web Token (JWT) [151]; some other proposals [85] uses a special token format which allows for additional features. All the proposals communicate through a RESTful API.

Another framework for securing API-enabled IoT devices in smart buildings [118] is also inspired by OAuth and uses JWT. The proposed security manager is split into two services to enable better scalability. The first service is an authentication manager that authenticates users or services with a process similar but not identical to OAuth and issues a JWT. The second service is an access control manager that verifies whether the access is allowed, based on XACML rules set by the system administrator and the requesting side’s identity (which is provided by the token).

Also, Alkhresheh et al. build their framework [110] around XACML policies.

Their framework eases maintenance and increases security by generating XACML policies based on the attributes, context, and predication. The policies are then continuously enforced. Administrators of the system describe the policies in the elementary format, consisting of simple policies that together form more complex ones and are used to generate XACML policy dynamically.

Blockchain technology is used in [115] to store, distribute, and verify authorization rules. Every node in the network has a full database of all access control policies for each resource-requester pair in the form of transactions. Access is granted by giving a token to the requester entity and propagating it in the blockchain. The blockchain also serves as an auditing and logging tool. Trust in the network is based on the distributed nature and large size of that network; it is challenging to gain unauthorized access or disable the network by attacking a central element. A slightly different approach using blockchain is presented in [94]. Rules-based on OrBAC [152] are distributed through a blockchain, and based on the history of the communication; the rules are updated with reinforced learning algorithms. Another blockchain utilization is shown by [105]. The article describes cross-domain permission sharing and access control, which is currently done by a trusted third party or resource owner. The article introduces authentication and authorization sharing based on the blockchain that mitigates the single point of failure risk. The security rules are enforced by "smart contracts" that are either local, e.g., per domain or deployment, and single global that stores global security policies. As the blockchain principle is currently a trending research topic for the IoT, there are also other authorization framework proposals based on it [109], [122].

Tasali et al. [90] discusses current standards for healthcare devices, including Integrated Clinical Environment (ICE) [153] and Medical Application Platform (MAP) [154]. The conclusion is that they barely address authorization and authentication (if they address it at all). Their solution is based on ABAC, enhanced with attribute inheritance inspired by RBAC. Attribute inheritance allows the "plug-and-play" configuration of new devices based on device types represented as attributes pre-set on the devices.

Another option is to isolate each function of the device and provide access just to that functionality [90]. Functionalities are slightly similar to the concept of microservices. The proposed functionality-centric access control framework mainly reduces application-level attacks on "Misused functionality" or "Reduced functionality".

Djilali et at. [124] builds on top of RBAC authorization system that assigns users and devices into teams. They are one-off collaboration units and are created

Article	Centralized	Decentralized	U2M	M2M	Context-aware	Specifics
[80]	Yes	Yes	N/A	N/A	No	Rules tied to the data
[82]	No	Yes	Yes	Yes	No	Constrained devices
[83]	Yes	No	No	Yes	No	ARM compliant; describes access control generally
[85]	Yes	No	Yes	Yes	No	OAuth; tokens
[88]	N/A	N/A	Yes	No	Yes	Biometric information used
[90]	Yes	No	Yes	Yes	Yes	Supports with attribute inheritance
[91]	N/A	N/A	No	Yes	No	WS-Security adaptation for IoT
[93]	Yes	No	Yes	Yes	Yes	Full security framework
[94]	No	Yes	Yes	Yes	No	Reinforced learning to update rules
[95]	N/A	N/A	Yes	Yes	Yes	Perception layer framework
[97]	Yes	No	Yes	No	No	Smart home
[100]	Yes	No	Yes	Yes	No	OAuth
[101]	Yes	No	Yes	Yes	No	Functionality based
[102]	No	Yes	Yes	Yes	Yes	Continuous context-aware authorization
[105]	Yes	Yes	Yes	Yes	No	Blockchain; policies sharing
[107]	No	Yes	Yes	Yes	No	Blockchain
[109]	No	Yes	Yes	Yes	No	Blockchain
[110]	Yes	No	Yes	Yes	Yes	XACML
[111]	Yes	Yes	Yes	Yes	No	ARM compliant; ABAC; trust based
[112]	No	Yes	Yes	No	No	Tokens; Possible to share permissions
[113]	No	Yes	N/A	N/A	Yes	Data decryption only with correct attributes
[114]	No	Yes	N/A	Yes	Yes	Bayesian decision theory for authorization
[115]	No	Yes	Yes	Yes	No	Propagation through blockchain
[116]	Yes	No	Yes	Yes	No	OAuth; tokens
[117]	Yes	Yes	Yes	Yes	No	Access control specified for functionalities
[118]	Yes	Yes	No	Yes	No	OAuth; XACML; tokens
[119]	N/A	N/A	No	Yes	No	Constrained devices
[120]	Yes	Yes	No	Yes	No	Gateway, device and cloud share data encryption
[121]	No	Yes	Yes	Yes	Yes	User centric; smart power grid
[122]	No	Yes	Yes	Yes	No	Blockchain; capabilities
[123]	No	Yes	Yes	Yes	Yes	Continuous trust verification
[124]	Yes	No	Yes	Yes	Yes	RBAC; teams

Table 3.6: Summary of authorization articles

ad-hoc and last only when the collaboration is needed. The security rules are enforced by a central server that has access to global and team context.

A proposal for energy-constrained devices called Time Division Multiple Access is described in [119]. The schema is well suited for sensors with known communication patterns, such as a repeating communication schedule in which sensors periodically report data. The proposed communication scheme optimizes the trade-off between device lifetime and distortion of the data transmitted. Another different application of ABAC focused on reducing storage and communication overhead is described in [95].

Sicari et al. provides a full specification for a security framework for smart healthcare [93]. It describes three main points (locations) for policy enforcement – a policy administration point, a policy enforcement point, and a policy decision point. The access roles are described using XML in a format inspired by ABAC. Different domain-specific article [121] describes user-centric IoT platform to empower users for managing security and privacy concerns in the Internet of Energy. There exists also solution specific for a smart home environment [97] that extends FIDO [155]. A user on his phone needs to authorize all the device's actions. When the user acquires a new device, he needs to register it and provide authorization attributes and for this uses registration token issues by manufactured and provided with the device.

Another access control model for IoT running in the cloud [120] secures data using hierarchical attribute-based encryption. The encryption is done in two steps. The first part of encryption is done on the device; the secondary encryption is done on the gateway. This reduces the load on the device. Decryption is likewise split between the cloud and the device in order to save application resources. The encryption scheme's hierarchical nature allows updating security policies using an update key based on information from the data source, without the device itself needing to re-encrypt the data.

Three of the reviewed papers [88], [91], [107], [123] discuss authorization only tangentially. The complete overview of authorization research can be seen in Table 3.6.

3.2.3 Services

This section presents an overview of the solutions that either support IoT-as-a-service or provide security-as-a-service. This means that at a minimum, the security client (an entity) or security provider follows the principles of SOA [A.13]. Frequently, both of the actors can be viewed as services. In this research review, I have 16 research publications that include SOA compatibility, although not every paper in this category uses the term SOA or "service"; instead, they are frequently called by synonyms, e.g., "central entity", "authorization or authentication server". The majority of the centralized security approaches can be viewed as a service.

Most of the surveyed proposals contain an identity management, authentication, or authorization service. An application in the IoT environment may offload the authentication process to such a security service [78], [79], [83], [85], [89], [93], [96], [98], [100], [106], [108], [126], [129]. A few proposals also allow the distribution of access rights or other properties used for authorization from the service to its clients [77], [83], [121], [125]. Some of the services also provide additional features like enhanced user privacy [77], [87], [102], [125]. They anonymize entity identities by hiding identity details from the service provider, and guarantee the entity's identity by the trustworthiness of the identity management service itself.

Two of the papers in this category stand out. The first adapts the Web Service (WS) Security specification [156], which is intended for loosely-coupled distributed systems, to the IoT environment by extending it to allow identity management functions to be offloaded from computationally "weak" devices to "strong" ones [91]. The proposed method, termed DPWSec, also simplifies the original WS-Security specification by removing unneeded portions: multi-hop security, statelessness, hosting and hosted devices, and the device profile communication model. The second paper describes a security framework within the scope of the Device Profile for Web Services using the XACML standard for rule description [128]. It describes three parts of the framework – the policy enforcement point (where the policies are enforced), policy decision points (where the policies are evaluated), and policy information points (where the audit logs are kept).

3.2.4 Identity Management

Identity can be viewed as a set of user attributes, both virtual or real. Identity management is the mechanism of storing and retrieving user identities. Typically, users are forced to have more unconnected identities for various services. In the IoT environment, the identity should be available for the whole IoT network (or at least some significant part) while preserving the user's privacy, although it does not mean that each user must have a single identity. The identity concept is also extended from users to include sensor identities in the IoT. Identity management is closely connected to authentication, which verifies that a user (or a device) is the owner of that identity, and authorization, which is the process of granting access to a resource based on user attributes (i.e., identity). Eight of the articles address identity or identity management at least partially.

Traditionally, user identity contains the principal along with credentials used for authentication. This renders a privacy risk, especially if the identity is shared with multiple services whose operators are not known in advance, and that might appear on and disappear from the network at any time in the dynamic IoT concept. Many of the articles tackle the problem of privacy by limiting a user's identity to only their attributes, without any unique information that could lead to the disclosure of their identity. One of the proposals is for a trusted party to issue cryptographic containers containing user attributes [77]. It is not specified that the trusted party must be a single entity in a network, so we can assume that multiple trusted parties can exist simultaneously. Also [130] proposes using attributes instead of identity for authorization. Gusmeroli et al. propose a slightly different approach using capabilities instead of attributes [125]. This proposal also supports anonymous capabilities that allow authentication without disclosing identity. Fremantle et al. described federated identity model [129] based on OAuth 2.

The problem of assigning an identity to devices is described in [126]. An IoT device inherits its user's identity through various methods based on a relationship between the user and the device. They formulate methods for devices strictly connected to a single user and identity extensions from users to devices that frequently change users.

A complete framework for decentralized identity management to enhance user privacy is introduced in [96]. It defines partial identities as the least sufficient subsets of full identities for a requesting service that does not disclose any unnecessary information about a user. A different decentralized identity management framework [135] takes the device's trust into the context. The trust is dynamically calculated based on the history of interactions and the trust of the participants. There is also a very similar concept [103] with confidence that is calculated from contextual information.

The principle of storing a user's biometric information in access points, serving like identity servers, and thus linking a real user's identity with his virtual identity through wearables is described in [84]. Some articles suggest to move the identity management part into blockchain network [107], [108], [122]. The rest of the articles [80], [95], [99], [101], [102], [110] deal with the identity management only partially and the main contribution of their work lies in other areas.

3.3 Context awareness

One trend in contemporary application development is a movement towards context-awareness. Context is defined by Abowd [5] as any information that can be used to characterize an entity's situation. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves. In the context of the IoT it can be extended to not only interaction between a user and an application but also between two applications.

Solutions using context-aware security can provide a much better user experience as well as increased security [67]; often, both can be achieved at the same time. Nevertheless, the level of interest in context-awareness from a security perspective has not reached the same level as interest from the user experience perspective, likely because computer security is traditionally a more conservative computer science domain. In this section, we focus on research that does speak to an interest in context-aware security.

The most common approach to achieve context-aware security is using ABAC. It differs from RBAC in that an entity (a user or a device) performing an action is not authorized based on matching the roles it is assigned to roles that allow specific actions. In ABAC, every action is mapped to a specific set of attributes an entity must possess in order to take that action. An example of such a rule for reading a document is that the entity must be from the same department as the creator of the document, must be employed in a management position, and must be located in the same building or complex.

One option is to specify access rules using ABAC for every piece of data at creation time and join those rules with the data so that during network transportation, updates, or copying, the rules stay consistent. In order to manipulate the data, an entity must possess the specified attributes [113]. Another method is to use a three-module architecture. The first module, a policy enforcement point, is responsible for invoking checks on access rules. The second, a policy information point, gathers information about an entity's attributes, including their context. Finally, a policy decision point compares security rules with the information gathered about the entity and decides whether the action is allowed or declined [90], [132]. Security rules can be written in XML using XACML [90], [110] or using the Ontology Web Language [132]. While [93], [95], [121], [127], [130] and [124] do not mention context information specifically, the ABAC implementations in those papers could also utilize context-aware attributes.

Instead of extending ABAC, another option is to adapt the well-described Capability-based Access Control (CBAC) [157] architecture. A capability (known in some systems as a key) is a communicable, unforgeable token of authority. It refers to a value that references an object along with an associated set of access rights. This token may contain additional contextual rules, defined in XACML format, which must be satisfied for the token to be valid [125]. Variation on this is using Distributed Capability-Based Access Control [83] as described in [96].

A novel authorization architecture based on Bayesian decision theory [114] also considers context. The trust parameters of history, knowledge, and reputation (described in the Authorization section) may include contextual elements that are acquired either directly by the device itself or provided indirectly by a peer device. Machine learning techniques used to enhance access rights [94] also consider the context in terms of a history of the previous interaction.

Biometric information may be considered “contextual” by definition, so biometric authorization is context-aware [84], [88], [99], [104]. Many devices, especially wearables, directly measure the user's physical traits (e.g., heart rhythm or body temperature). Other "nearable" devices can provide additional information such as weight or gait, both of which can be measured by video sensors. All of this information can be compared to a user's known physical or kinesiological properties.

Beyond simple biometric data, a user's digital life may be considered as a context for identity management. A user's photos, videos, blog posts, and browsing history can be used to authenticate that user [89]. Given sufficient digital history, security questions can be devised which no one but the authentic user can answer. This benefits the user by not needing to memorize passwords or carry other credential material; their own memories are sufficient. Another similar proposal, which restricts context to information from network traffic, authenticates using contextual information provided by a smart home [131].

A different approach is to evaluate the history of the actions. It can include communication patterns, actions performed, or even a typical context in the given time for a user or a device. For sensors, the values they produce can be observed, and some patterns or limits can be determined [123]. Then it is used as an additional factor for authentication. A similar approach can be used on users. IoT devices can monitor their activity, and the usual patterns can be evaluated for authentication [102]. Alternatively, communication history can be evaluated - based on the participants and their trust in the given moment, the device's current trust might be calculated [103], [136].

3.4 Existing vs. novel approaches

Existing research projects in IoT security that propose an actual solution or method can be roughly aligned to two categories: those which extend or adjust existing architectures or programs to better suit the IoT environment, and those which propose entirely new ideas to solve environment-specific problems. However, the classification is not strictly binary, and it is often difficult to judge the novelty of any particular proposal. The reader will note that all research is meant to be "novel"; we use the word here in a narrower sense to mean an entirely new approach that does not make use of existing technologies or standards.

The works we considered that apply or adapt existing technologies and methods from other security domains to the IoT environment often consider OAuth 2 technology [81], [86], [100], [106], [116], [118], [129]. Two proposals also adopts the WS-Security specification to IoT devices and communication between them [91],

[128].

The most innovative solutions share some common properties. Most of them are suitable for distributed use, and none require administrator interaction. They can handle device connection and disconnection, as well as security rule distribution and validation. Often the responsibility for the creation of access rules is moved from administrators to data owners. Some papers show operation with trust between devices and dynamic calculation of trust among various communication partners [103], [111], [114], [130], [135]. One proposal adjusts ABAC to be more dynamic and allow a device to pick its own attributes; other devices must subsequently confirm that the device really does possess the claimed attribute. Security rules are set during data creation using ABAC and then connected to those data for its whole life-cycle [127]. Another innovative approaches suggest propagating all security rules through a blockchain in the network [94], [105], [107]–[109], [115], [122]–[124]. One of the researches proposes access control based not on roles or attributes, but rather on functionalities of the IoT node[117]. Access control for cloud applications based on attributes [120] using the computational power of sensor gateways the cloud itself is suitable for constrained devices.

In summary, there are various novel proposals [94], [103], [111], [114], [115], [117], [120], [127], [130], [135], especially focusing on distributed solutions [94], [105], [107]–[109], [114], [115], [120], [122]–[124], [127], [130], that potentially suit the IoT environment better in terms of scalability, maintainability, and flexibility, but due to their novelty it is difficult or impossible to predict which ideas might be adopted or see wide use. A significant amount of research **ieee_ja**, [81], [85], [91], [95], [100], [106], [116], [118], [128], [129] is focused on adoption of existing technologies; all exhibit promising results.

3.5 Distribution vs. centralization

The IoT is a diverse, complex, and fast-changing environment. It comprises a large number of devices that interact autonomously. Objects also appear and disappear autonomously and with high frequency. Given these differences from a more standard network environment, we focus in this section on what paradigms are used in the security solutions.

A conventional, centralized approach is straightforward to set up, maintain, and audit for system administrators. It also presents a stable point in the network from which users and applications can build trust. Implementing centralized solutions is simpler both for the central server as well as for applications using it. Many of

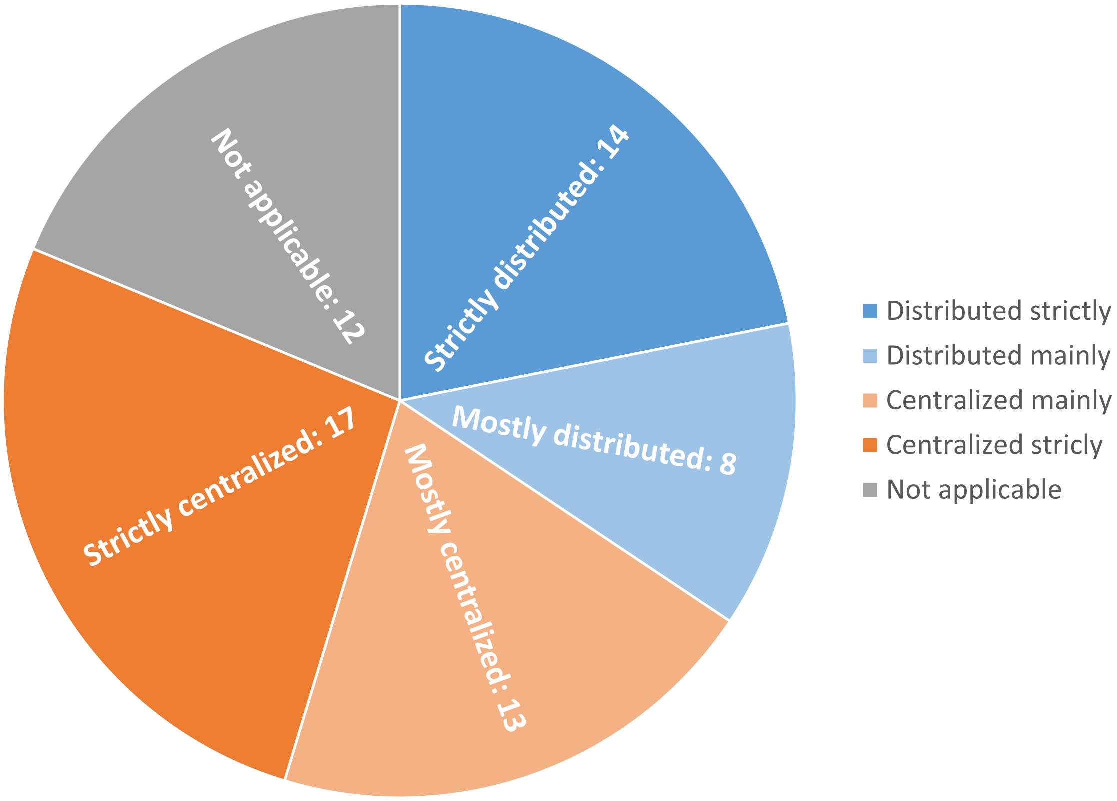

Figure 3.4: Categorization of distributed and centralized solutions

the existing centralized solutions for networks and application can be extended to operate in the IoT environment without overly costly adjustments. However, using a centralized architecture in the IoT presents several drawbacks, including limited flexibility and scalability.

By contrast, the attributes of distributed architectures are entirely opposite. They scale well and are built with flexibility as the primary goal. However, synchronization, maintenance, and auditing present serious difficulties. There is also the issue that no single trusted central entity stands behind them, which may be required by business users, legal entities, or others.

To further complicate matters, the line between distributed and centralized solution is often not clear. While some solutions can be considered exclusively in one category, a significant number of proposals may work under both paradigms. Figure 3.4 shows a chart of distributed and centralized solutions.

Requiring a central server for identity management prevents distributed operation for obvious reasons. Sometimes this limitation is imposed for domain-specific reasons (e.g. in the healthcare domain [80], [84], [93], [116], [121], [124]); other times it arises simply as a function of the technologies or methods employed [81],

[85], [97]–[100], [104], [129], [131], [134]. In one proposal, the authentication method requires having as much historical data about an entity as possible, to the point that authentication data storage requirements make it impractical to host such data at multiple locations [89].

At the other end of the spectrum, the technologies used in some proposals specifically preclude centralization. For instance, methods which rely on the creators of data to specify security rules, or which grant access selectively, do not operate with a central server [82], [96], [106], [112], [114], [121], [123], [127], [137]. Blockchain-based access rule verification [94], [107]–[109], [115], [122] also can not be centralized, and the same applies to extensions of the ABAC system which rely on peer devices to confirm an entity's attributes over the network [130].

Most of the ideas in the papers surveyed can be used in both centralized and decentralized architectures. Centralized solutions can be often decentralized by multiplying central elements [76]–[78], [83], [87], [90], [101]–[103], [105], [111], [117], [118], and decentralized proposals can be centralized by limiting the number of security control elements to single node [79], [113], [120], [125], [130], [136]. Similarly, some of the research we reviewed [88], [91], [92], [95], [110], [111], [119], [128], [133]–[135], [138] cannot be categorized in either category. They work equally well for either architecture without modification and can be seen as complementary extensions for complex security solutions, helping with particular issues (e.g. authentication, auditing, context awareness).

3.6 User vs device-centrism

In IoT two basic communication patterns exist: either user interacts with devices, or devices interact among themselves. The first type is designated U2M category. The other scheme of communication is designated M2M. Some of the proposals fit both patterns; this section explains how the security models support particular communication models and their limitations.

One important restrictive factor is the need for human input to the interaction. In some cases, various information about the actual user is required for security reasons: biometric information [84], [88], [98], [99], [104], [138], a user's digital history [89] or real world history [102], or a user's location [76]. Other approaches require direct user interaction such as scanning QR codes, providing input on a phone device or generating password using words [79], [92], [97]. Any of these cases requires U2M communication.

Generally a device is capable of constant and repetitive tasks, but its decision

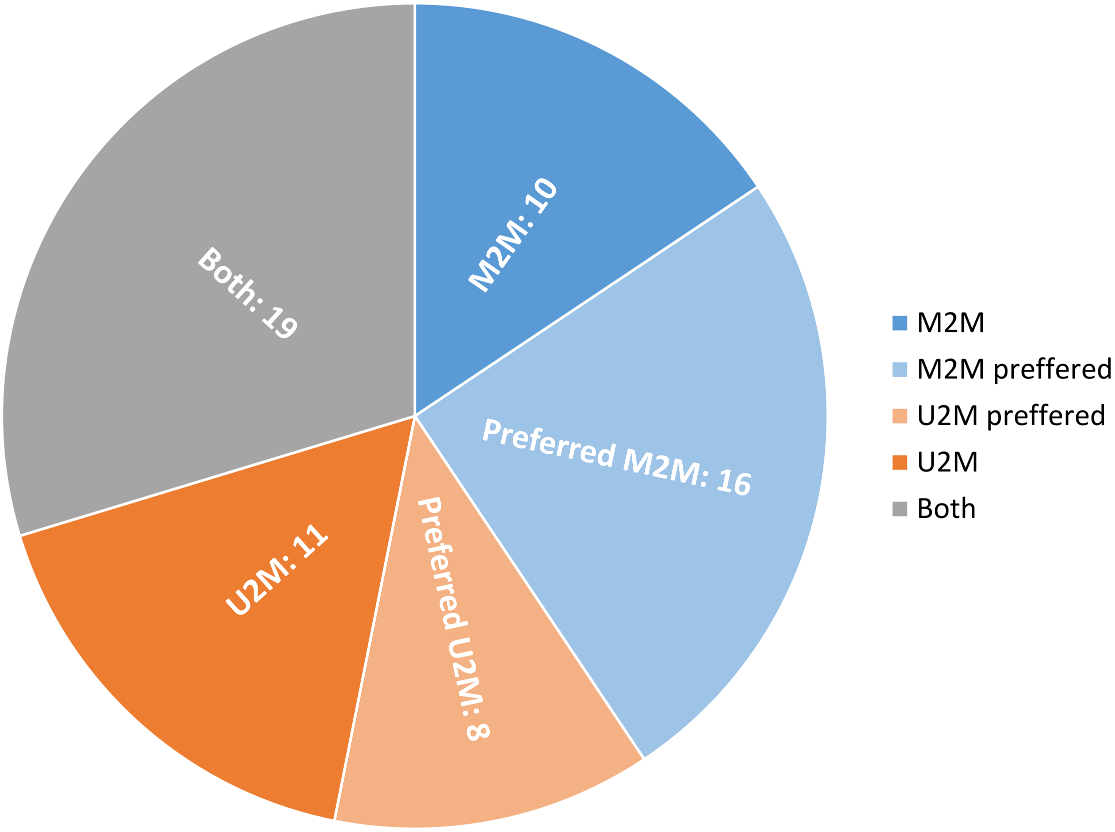

Figure 3.5: Categorization of U2M and M2M solutions

capabilities are limited: goals or objectives can only be set by a user. Users, on the other hand, may find monotonous or continual-load requirements onerous at best and impossible at worst. Given these differences in capability, the adaptation of existing M2M security technologies [83], [91], [95], [118], [135], [137] works well for IoT scenarios where a user is not required. Proposals exist for M2M authentication even with low-resource devices [82], [119], [120], [123].

Finally, many of the solutions described in U2M research can be used for M2M identity management with little to no modification [76], [80], [86], [93], [96], [103], [112], [132] and vice versa [77], [90], [94], [105], [107], [111], [114], [115], [117], [121], [122], [128], [130]. Some of the research even includes existing U2M technologies being used for M2M purposes [85], [116], and many of the papers surveyed are useful for either communication model [78], [81], [87], [100], [101], [106], [108]–[110], [113], [124]–[127], [129], [131], [133], [134], [136].

Figure 3.5 shows that research contributions in the U2M communication model occur with similar frequency to those in the M2M model. The vast majority of projects can be used for either communication scheme, which demonstrates the versatility of the security solutions and proposals.

3.7 Threats to validity

A literate overview is a highly subjective type of research and therefore suffers from threats to validity. I have identified several threats that need to be addressed or at least mentioned. In order to eliminate most of them, I have followed recommended guidelines for conducting systematic studies [68].

The evidence selection is based on professional indexing sites. I could miss some articles published in other sources (e.g., journals not indexed in WoS). Also, the queries I use to search for articles explore only abstracts. This means that articles that should have been included may not have been because they contained some of the excluded words or did not contain any of the included words. I tried to eliminate this by testing our queries against the manually-selected control set.

Data extraction bias is another possible threat to validity. I addressed this primarily by ensuring that each paper received several individual reviews focused on each research question. Using the RAKE algorithm to extract paper keywords also mitigated data extraction bias somewhat because the same extraction method was applied to each paper, apart from any human factors.

Data were acquired at two different points in the time, in 2017 and 2020. Also, they were processed with a gap of three years. In that time, my subjective view on the articles could have changed, and therefore the selection either by reading abstract (or the whole articles) could slightly evolve. Also, as noted, the second time the review was done, only WoS SCIE was examined, which might leave some significant paper unnoticed. However, I believe that every significant research is published in a journal indexed by WoS.

Exclusion and inclusion of the papers due to their scope is also a potential threat. To mitigate this threat, I followed methods for the selection criteria suggested in [68]. I have read numerous related works and spent considerable time reading the selected papers to ensure they fit within our considered scope. I removed papers that focus specifically on cryptography, networking, and low-level device security. I have also excluded papers that do not provide specific results, that list only suggestions or opinions without solution proposals.

All of the papers were treated equally in the survey, although not all published research has the same quality or impact on the community. I provide some overview of each article's impact in Table 3.7 and Table 3.8, including metadata about the impact of the paper and possible quality of the publication source. To measure community impact, I have chosen two sources: data from publishers and Google Scholar [158]. Publishers generally provide their own list of citing works. One disadvantage of using this publisher-provided data is that it may often miss citations from sources unknown to it. Therefore, Google Scholar was chosen as a universal, most fully populated article aggregator. It provides its own citations list, but they also include self-citations, and it may take up to few months for articles or citations to appear there. To quantify the quality of the publishing media, I chose two methods. For journals, I use Impact Factor [159] from WoS SCIE (2019) as it is the most prominent and possibly oldest journal indexing tool. Ranking conferences proves to be more difficult. The most appropriate measure for our needs seems to be the latest 2020 Computing Research Education (CORE) Association of Australasia conference ranking [160] as it presents independent rankings of conferences with any sponsor. It ranks conferences with letters C, B, A, and A* for their quality (A* is the best, C is the worst). A disadvantage is that not all conferences are included in the ranking, and the ranking itself is managed by a small group of scientists from a particular geographic area. The citation numbers were updated on February 7th.

Article	Published in	IF or CORE	Year	Source citations	Google citations	Views
[76]	Conference	N/A	2016	6	19	437
[77]	Conference	N/A	2016	16	23	526
[78]	Conference	N/A	2015	7	43	586
[79]	Conference	N/A	2016	3	6	898
[80]	Journal	2.892	2017	9	22	997
[81]	Conference	A	2015	7	22	1400
[82]	Book chapter	N/A	2015	2	2	13
[83]	Journal	11.42	2015	82	117	2400
[84]	Conference	N/A	2017	10	32	629
[85]	Journal	1.151	2017	2	6	1826
[86]	Conference	B	2014	0	1	1400
[87]	Conference	N/A	2016	4	3	621
[88]	Journal	4.231	2017	32	47	3489
[89]	Conference	C	2015	6	13	275
[90]	Conference	C	2017	2	9	365
[91]	Conference	N/A	2015	2	5	221
[92]	Conference	A*	2016	1	3	337
[93]	Journal	N/A	2017	11	30	85
[94]	Journal	N/A	2017	32	94	N/A
[95]	Journal	N/A	2014	97	88	N/A
[96]	Journal	1.508	2017	25	38	886
[97]	Journal	13.727	2018	26	76	216
[98]	Journal	2.645	2018	3	4	895
[99]	Journal	9.936	2018	11	27	1066
[100]	Journal	2.645	2019	5	7	1281
[101]	Journal	13.727	2019	5	7	48
[102]	Journal	2.645	2019	5	6	1327
[103]	Journal	9.936	2019	1	4	363
[104]	Journal	3.745	2020	1	1	480
[105]	Journal	3.745	2020	2	1	1000
[106]	Journal	1.151	2020	0	0	852
[107]	Journal	3.458	2020	15	28	1722

Table 3.7: Community impact of articles. Part 1/2.

Article	Published in	IF or CORE	Year	Source citations	Google citations	Views
[108]	Journal	0.648	2020	0	0	N/A
[109]	Journal	1.061	2020	0	0	121
[110]	Journal	9.936	2020	0	1	217
[111]	Journal	3.05	2016	67	112	1829
[112]	Conference	N/A	2015	6	8	297
[113]	Conference	C	2015	6	11	235
[114]	Conference	N/A	2015	6	9	413
[115]	Conference	N/A	2017	92	235	5100
[116]	Conference	N/A	2016	10	23	986
[117]	Conference	C	2017	13	30	351
[118]	Conference	B	2016	8	17	458
[119]	Journal	6.779	2017	7	14	652
[120]	Journal	2.892	2017	19	28	855
[121]	Journal	1.151	2017	3	8	1776
[122]	Journal	9.112	2020	2	5	436
[123]	Journal	3.275	2020	1	1	1180
[124]	Journal	1.594	2020	0	0	N/A
[125]	Journal	1.366	2013	185	291	240
[126]	Conference	N/A	2017	2	4	1000
[127]	Conference	N/A	2016	1	4	1600
[128]	Conference	N/A	2014	11	19	129
[129]	Journal	1.546	2018	3	11	1938
[130]	Journal	11.051	2017	20	37	1141
[131]	Conference	B	2015	88	241	5886
[132]	Conference	N/A	2014	3	8	395
[133]	Conference	B	2017	11	31	643
[134]	Conference	C	2015	3	20	406
[135]	Journal	2.024	2018	1	11	199
[136]	Journal	2.602	2018	6	7	244
[137]	Conference	N/A	2017	0	1	790
[138]	Conference	N/A	2016	4	9	464

Table 3.8: Community impact of articles. Part 2/2.

Chapter 4

Context retrieval and Authentication

Obtaining the context is the initial and crucial part of the context-aware applications. The contextual information is used for context-aware security, and without coherent, relevant, and up-to-date data, the context-aware security may not provide valid results.

Traditional use cases allow obtaining of the context data from a single place (sensor) and then distributing it to the other interested participants is enough. However, we have to get contextual information from as many participants as possible for security usage, preferably from all of them. This information also needs to be in a unified format so the security rules can be reused across the devices and applications.

In this chapter, I present part of the research that focuses on minding novel contextual data that could be accessible to all IoT devices. It explains the proposed method, describes the algorithm, and then it evaluates the solution both in real-world scenario and simulation.

To demonstrate the value of context usage, I use it as an additional authentication factor. The traditional authentication is not modified; only the additional factor (something I am) is added. It is based purely on contextual information. The contextual information could also be used for context-aware authorization, as it is described in the following chapters.

The achievements of the presented research can be summarized as follows:

- Presentation of an additional authentication factor for usage in the IoT environment.
- Illustrating how to set up our method in a network.
- Discussing how various settings affect the proposed method and how to determine the ideal values.

- Demonstration of the method's feasibility based on a network with hundreds of unique devices and providing experimental data allowing better insight into and applicability of the method.

The research described in this chapter has been initially explored in the conference paper [A.6], following another conference article with progress [A.7] and final journal article [A.1] has emerged from the research effort.

4.1 Proposed method

The idea of the proposed method is based on the regular network context reports provided by every IoT device. They retrieve a list of all devices discoverable in the network and send it to the server regularly. Ideally, that information is passed along during every server request. Due to the network bandwidth, storage, computation capabilities of the server, and other limiting factors, it can be restricted to a specific reasonable time frame (e.g., every 15 minutes) to reduce communication overhead. The server subsequently stores the data for further use, evaluates the received data, and eventually proceeds with further actions. Such actions may include an additional authentication request to the suspicious device (which may or may not be the device that triggered the action), a notification to a network administrator, or even a limitation to or removing network access for the suspicious device. A network context scan is performed on end devices, and the server performs only a context evaluation, which results in great scalability.

The utilization of our approach and its full possibilities requires a significant amount of contextual data gathered over extended periods of time, preferably in various distinct physical locations, across multiple different networks, and mainly with the knowledge of the security incidents that happened. Given such an extensive data set, it can be analyzed using standard algorithms based on decision tree induction [161] or advanced adaptive fuzzy rule-based classification [162]. Once the patterns are recognized, they can be searched for in real-time, and appropriate control mechanisms can be activated as needed. Unfortunately, I do not possess such a data set. Therefore, in this chapter, I describe a method to analyze a particular device's network context.

The method utilizes "recurring" devices for analyzing network context. A recurring device is a device that has been in the network for several consecutive days. For example, such a device is typically present in the network at a particular time. The Internet follows standard OSI networking model [163], possibly with

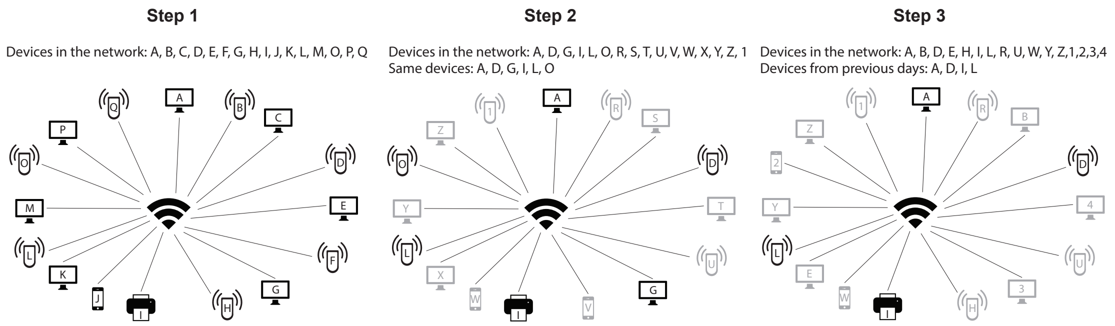
Step 1
Devices in the network: A, B, C, D, E, F, G, H, I, J, K, L, M, O, P, Q
Step 2
Devices in the network: A, D, G, I, L, O, R, S, T, U, V, W, X, Y, Z, 1
Same devices: A, D, G, I, L, O
Step 3
Devices in the network: A, B, D, E, H, I, L, R, U, W, Y, Z,1,2,3,4
Devices from previous days: A, D, I, L

Figure 4.1: Creation of recurring devices set in three steps

MAC layer protocols adapted for IoT devices [164]. Therefore MAC addresses are used as device identifiers because, by definition, they are unique. Their potential counterfeit problem is not significant in most of the scenarios as multiple devices with spoofed MAC addresses would need to be introduced into the network. The possibility of the attacking device changing its MAC address does not affect our method more than any other device with a spoofed MAC address, as this MAC address is treated as one of the addresses on the network. A potential successful attack targeting our method would lead to a higher ratio of false positives, which would not affect user experience or security (compared to not using our method as an additional factor at all).

The recurring device list is created specifically for a given network. While a recurring device may be a recurring device in more than one network, this is rarely the case.

4.1.1 Illustration of The Proposed Approach

An example of such a situation is a personal device carried by a user; the device is in a network during the day when the user is at work but in a different "home" network at night.

Recurring devices are determined based on historical values that are stored by the server. If a device appears in the network at the same time over multiple consecutive days, it is marked as a recurring device. Recurring devices are determined from a limited historical time frame (e.g., the last five days), and therefore, the set of recurring devices can vary from one day to the next. When the process is started, recurring devices cannot be determined, as there is no reference point. A list of recurring devices can be made when the time frame passes (e.g., five days). Recurring devices are calculated every day given the historical values. The algorithm takes all devices from the first day and marks them as candidates. Every

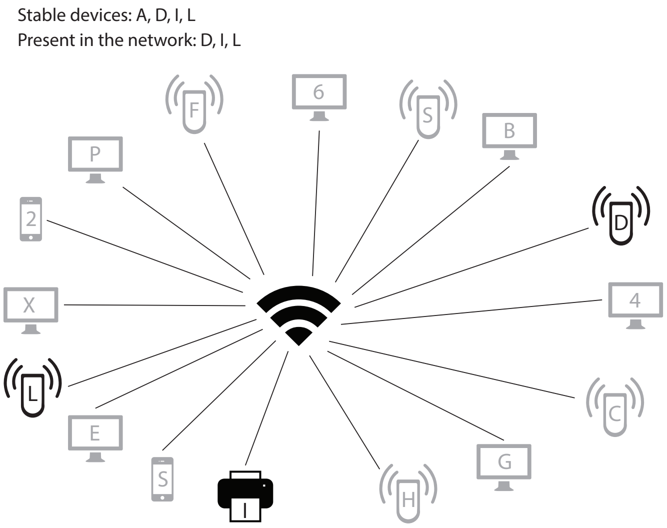

Figure 4.2: Using network context to determine changes in the network

subsequent day, it removes devices that are not present during the day from the candidates' list. After all the steps are completed, the candidate list is the final list of recurring devices.

Figure 4.1 illustrates the three-step determination of recurring devices. The steps illustrate a network at the same time over three consecutive days. The sample network consists of 16 various devices; thus, I can easily visualize it. Real networks often contain hundreds of network elements. During step 1, all devices are considered recurring device candidates. In step 2, there are the same six devices as in step 1. These are new recurring device candidates. In the final step, four devices from the candidate list are present. This list is a new list of recurring devices and can be used on the following day. Each step represents a single day. After the first step, we cannot determine recurring devices because there is no reference point. On day two, I make a list of candidates with the devices that have been active during both days; on the following day, the list of candidates is reduced again. If the number of previous records is larger than that from the time frame used for determining a recurring device, then some devices can also be added.

During communication, a device sends the list of all reachable devices in the network. The same rules described above are applied to determine recurring devices; thus, the device does not need to obtain the list for every request. The server compares the sent list with the list of recurring devices (which I call the benchmark) for the given network for a roughly similar time frame. It also uses the provided list to modify and verify the benchmark for the following days. Our preliminary implementation of the approach can configure the desired recurring device match with the devices in the network. Figure 4.2 illustrates a network with 16 devices and a set of recurring devices consisting of four devices from the previous figure - A, D, I, and L. In this example, the match is 75% (device A is missing). If the threshold is not met (e.g., 70% match), then the network context of the device is marked as suspicious, and further steps can be taken–the administrator is notified, an additional authentication factor can be invoked, or a more sophisticated network search for malicious devices can be triggered.

4.1.2 Problem Model and Algorithm

We model the analyzed network as a set of devices $N = \{n_1, n_2, \ldots, n_n\}$, where device n is every network element with MAC address. Timeframe $t = (t_{start}, t_{end}), t_{end} - t_{start} < 1\ day$, is a time period during a single day. Times t_{start} and t_{end} can be equal; in such a case, the timeframe t is not an interval, but a time point. Age (denoted as age) is a number of consecutive days, for which the benchmark is created. I denote the day in which the analysis is performed as d.

Benchmark is $B(t, d, age) = \bigcap_{x=d-age-1}^{d-1} devices(N, t, x)$ where $devices(N, t, x)$ denotes set of devices present in the network in a randomly selected time from the timeframe t during the day x.

We define $match(t, d, age) = \frac{B(t,d,age) \cap N(t,d)}{B(t,d,age)}$ as ratio between number of devices in the benchmark and number of devices in the benchmark present on the network, where t is a timeframe and d is a day in which the analysis is performed.

Then, $Threshold$ is a value of $match(t, d, age)$ such that if $Threshold > match(t, d, age)$ the authentication check (as introduced in subsection 4.1.1) is passed.

In Algorithm 1 I describe the process to determine the $threshold$ and age. The algorithm accepts the following inputs:

1. Set of all analyzed timeframes T
2. Analyzed network N
3. Constant ε defining when to stop the algorithm
4. Constant lim which is the number of days for which I run the algorithm

The outputs of the algorithm are:

1. Age_{opt}, which denotes the optimal age
2. $Threshold$, which denotes maximal possible threshold for given lim

Algorithm 1: getAgeAndThreshold(T, N)

```
    Input  : Timeframes $T$, Network $N$, $\varepsilon$, $lim$
    Output : $Age_{opt}$, $Threshold$
1   $devices(N, t, d)$ = set of present devices in N for $t$ and $d$, $t \in T$, $d$ is day
2   $B(t, d, age) = \bigcap_{x=d-age-1}^{d-1} devices(N, t, x)$
3   $match(t, d, age) = \frac{B(t,d,age) \cap N(t,d)}{B(t,d,age)}$
4   $Age_{opt} \leftarrow 0$
5   $Match_{val} \leftarrow 1$
6   for $d = 3 \ldots lim$ do
7       for $age = d - 1 \ldots lim - 1$ do
8           $Match_{min} \leftarrow 1$
9           for each $t \in T$ do
10              $Match_{tmp} = match(t, d, age)$
11              if ( $Match_{min} > Match_{tmp}$ ) then
12                  $Match_{min} = Match_{tmp}$
13              end
14          end
15          if ( $Match_{min} > Match_{val}$ ) then
16              $\Delta Match = Match_{min} - Match_{val}$
17              $Match_{val} = Match_{min}$
18              $Age_{opt} = age$
19              if ( $\Delta Match < \varepsilon$ ) then
20                  goto 25
21              end
22          end
23      end
24  end
25  $Threshold \leftarrow 1$
26  for $d = Age_{opt} + 1 \ldots lim$ do
27      for each $t \in T$ do
28          $curMatch = match(t, d, Age_{opt})$
29          if ( $curMatch < threshold$ ) then
30              $Threshold = curMatch$
31          end
32      end
33  end
34  return $Age_{opt}, Threshold$
```

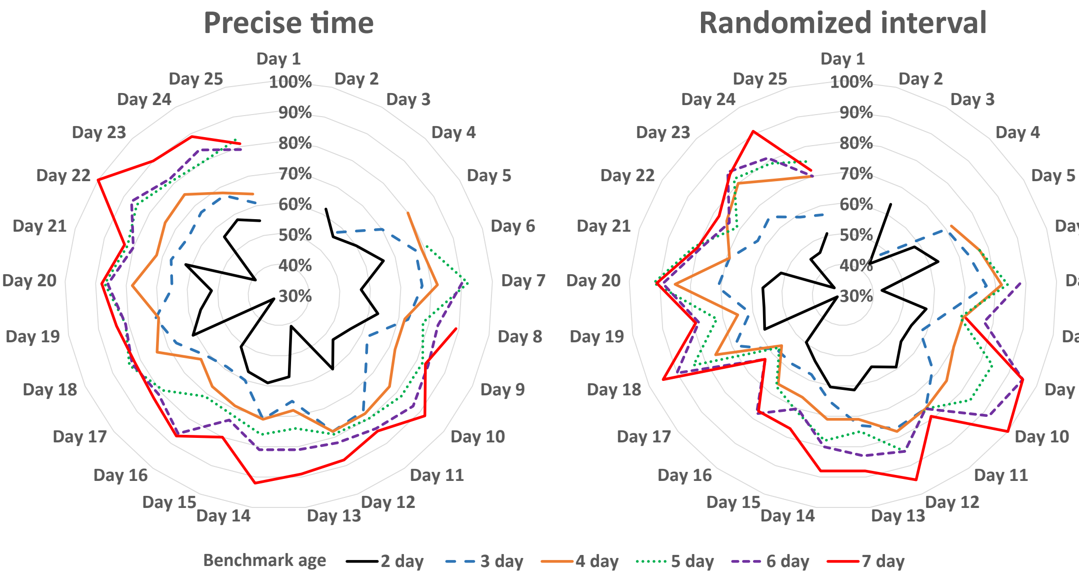

Figure 4.3: Percentage of recurring devices for every day using different benchmark age

The principle of the Algorithm Algorithm 1 is the following.

1. For network N create a set of benchmarks for a defined set of ages (2 to lim) (lines 6-24). During this process, two principal activities are conducted:

   a. Compare the last created benchmark with a previously determined best benchmark, which is the best value of the *match* function. If the latest benchmark has a better value of *match* function, consider this benchmark as the best one (line 15).

   b. When the value of *match* function of compared benchmarks starts converging to meet the algorithm stopping criteria defined by ε, return the best found *age*, denoted as Age_{opt}. (line 19)

2. For timespan from $Age_{opt} + 1$ to lim determine $Threshold$ such that all *match* for each of the analyzed timespans are equal or higher than $Threshold$ (lines 25 to 33)

4.2 Experimental Verification

4.2.1 Real-network evaluation

To verify the proposed method using a real network, I conducted a case study described in this section. To demonstrate the validity of the proposed approach, I

	8:00	12:00	16:00	Morning	Noon	Afternoon
Devices count	272	620	931	309	581	560
2 day benchmark size	71	128	156	90	128	138
2 day recurring devices	47	77	86	58	69	84
2 day recurring devices	66%	60%	55%	64%	54%	61%
3 day benchmark size	51	70	90	46	70	80
3 day recurring devices	36	54	65	32	49	57
3 day recurring devices	71%	77%	72%	70%	70%	71%
4 day benchmark size	41	50	67	38	46	58
4 day recurring devices	30	39	53	28	33	45
4 day recurring devices	73%	78%	79%	74%	72%	78%
5 day benchmark size	35	41	54	30	37	44
5 day recurring devices	28	33	44	24	26	36
5 day recurring devices	80%	80%	81%	80%	70%	82%
6 day benchmark size	31	31	35	26	29	30
6 day recurring devices	26	26	31	21	20	25
6 day recurring devices	84%	84%	89%	81%	69%	83%
7 day benchmark size	25	27	30	22	24	25
7 day recurring devices	21	23	26	17	18	21
7 day recurring devices	84%	85%	87%	77%	75%	84%

Table 4.1: Day 11 Benchmark age difference times: different benchmarks for specific date

performed: (1) evaluation using a real network and (2) simulation of the network with various possible events that could happen (e.g., recurring device disappearance or MAC address spoofing). Details are presented in the following subsections.

Initially, I determine relevant timeframes for a benchmark. Then, I determine whether the exact same time of the day needs to be used for the measurements during various days or whether an alternatively approximate interval can be used. Once I have such values, I proceed to determine a threshold for the percentage of recurring devices in the network based on historical network data.

We perform five weeks of measurement in the same network and conduct six control measurements. I have performed the case study on Baylor University Wi-Fi network in the Department of Computer Science with hundreds of unique devices. I choose this network for the experiment because it provides a considerable number of devices in which users periodically connect and disconnect (e.g., students' devices) with various schedules and devices that are always present (e.g., printers), and I was conducting the research during my visit there in cooperation with other researchers. I perform six analyses per day, evaluating the network only during weekdays. Three analyses are conducted at fixed times–08:00, 12:00, and 16:00 – and three are conducted at random times within specific time intervals representing morning (07:30-10:00), midday (11:00-13:00), and afternoon (14:00-17:00).

Initially, I aim to determine how many days are needed for the benchmark. I run the algorithm for 11 days. I run it twice - once for the fixed timeframes and once for the intervals. I show up to a 7-day benchmark for Day 11 in Table 4.1 with different benchmark periods. There is a gradual decrease in the benchmark size from over 100 devices in the two-day benchmark down to 25 devices in the 7-day benchmark. Theoretically, as these devices should be more stable, the percentage of recurring devices found increases, which is also generally the case on the example day. Note that for the last three days, the benchmark size and percentage do not vary considerably. This finding leads us to the conclusion that adding more than five days provides only limited benefits; thus, I choose five days as our benchmark period. Those findings are consistent across all measurement times, even for those taken randomly within an interval. I illustrate the percentage of recurring devices found for various benchmark periods for all days in Figure 4.3 where the 12:00 and midday measurements are used. The randomized interval measurements are illustrated on the right graph, and they fluctuate significantly more than the measurements taken every day at the same time, which are on the left graph. Note that only weekdays are used; thus, day 6 corresponds to a Monday. The algorithm yields a five-day benchmark that provides a percentage[1] nearly as good as that of the benchmarks consisting of a more extended period, with differences of only approximately 2% from the 6-day benchmark and 5% from the 7-day benchmark age, while also providing better stability than the 7-day benchmark.

[1] Comparing the minimal value from the 25 days

	8:00		12:00		16:00		Morning (07:30-10:00)		Noon (11:00-13:00)		Afternoon (14:00-17:00)	
Day	Devices	Recurring devices	Devices	Recurring devices	Devices	Recurring devices	Devices	Recurring devices	Devices	Recurring devices	Devices	Recurring devices
Day 1	343	N/A	769	N/A	729	N/A	568	N/A	599	N/A	568	N/A
Day 2	447	N/A	615	N/A	628	N/A	606	N/A	585	N/A	722	N/A
Day 3	349	N/A	645	N/A	753	N/A	337	N/A	629	N/A	781	N/A
Day 4	365	N/A	546	N/A	521	N/A	389	N/A	715	N/A	534	N/A
Day 5	245	N/A	456	N/A	557	N/A	191	N/A	537	N/A	580	N/A
Day 6	271	N/A	546	N/A	696	N/A	314	N/A	703	N/A	651	N/A
Day 7	429	96%	566	90%	573	76%	449	100%	518	88%	653	76%
Day 8	261	82%	691	83%	715	86%	182	69%	627	73%	705	82%
Day 9	416	100%	656	84%	514	82%	491	100%	578	89%	532	82%
Day 10	252	86%	540	87%	520	81%	280	96%	562	96%	446	76%
Day 11	272	84%	620	84%	931	89%	309	81%	581	69%	560	83%
Day 12	510	79%	772	82%	701	86%	198	79%	611	92%	577	83%
Day 13	316	79%	830	81%	981	82%	317	86%	765	83%	632	76%
Day 14	538	81%	689	81%	728	86%	162	79%	600	87%	746	88%
Day 15	320	77%	726	74%	709	78%	175	76%	677	73%	451	83%
Day 16	436	88%	811	86%	1166	76%	352	86%	600	84%	877	82%
Day 17	571	85%	765	81%	1004	79%	626	79%	720	66%	563	76%
Day 18	360	83%	1304	83%	1247	80%	351	88%	1206	83%	703	76%
Day 19	611	81%	938	81%	823	79%	549	81%	885	85%	664	75%
Day 20	304	82%	676	86%	739	81%	430	79%	966	89%	683	65%
Day 21	405	87%	1168	80%	928	73%	480	75%	1180	82%	829	77%
Day 22	564	79%	968	87%	723	77%	689	79%	994	81%	664	95%
Day 23	400	91%	818	82%	1147	78%	496	91%	1252	88%	986	72%
Day 24	602	92%	687	84%	708	75%	581	77%	740	81%	603	79%
Day 25	309	92%	576	79%	548	84%	378	78%	519	74%	386	75%

Table 4.2: Day overview of the 5-day benchmark: number of devices on the network and recurring devices for each measurement during every day.

	Devices count	Benchmark size	Recurring devices count	Benchmark match
8:00	272	35	28	80%
12:00	620	41	33	80%
16:00	931	54	44	81%
Morning	309	30	24	80%
Noon	581	37	26	70%
Afternoon	560	44	36	82%

Table 4.3: Day 11 measurement: devices on the network in the specific times and intervals with 5 day benchmark age

The next unknown piece is the difference between the measurements taken at strictly the same time and those taken during the same interval. Randomized measurements decrease the possibility of intentionally spoofing the network and providing fictitious MAC addresses to inflate the set of recurring devices. As in the previous paragraph, I use day 11 to demonstrate our findings. However, I now choose only a 5-day benchmark and illustrate the number of devices in the network, the benchmark size, the recurring device count, and the benchmark match for every time in Table 4.3. For every time or interval, I use the corresponding times or intervals on previous days to determine the benchmark. The table shows that there is a noticeable and randomly occurring decrease in the match percentage between the interval and corresponding fixed time measurement. I choose to continue the case study with fixed time measurements because they provide higher consistency. The measurements also confirm this higher consistency in Table 4.2, where the recurring devices for a specific time never drop below a 73% match, while the interval measurements can drop as low as a 65% match.

With the benchmark period set using fixed times for the measurement strategy, to run the control measurements, the only part that is missing is an optimal threshold for validating the network context. The algorithm Algorithm 1 gave us the output of 76% as the maximal threshold; I choose to lower it to 70% to give us some safety margin. Table 4.2 presents the network evaluation for every day and time or interval during our study using the five-day benchmark. Day 1 in the table corresponds to the Monday of the first week of the case study, with days 6, 11, 16, and 21 also being Mondays. The number of devices in the network varies from 250 to over 1000, with Fridays and parts of Monday being the days with the fewest devices and mornings being the time with the lowest number of active devices. However, the percentage of recurring devices is reasonably consistent,

Place	Devices	Bench.	Recurring devices	Percentage	Day
Supermarket	595	37	1	3%	6
Apartment	8	38	0	0%	9
Dinning hall	1095	38	0	0%	7
Commons	42	38	2	5%	9
Saturday	118	38	11	29%	10

Table 4.4: Control Measurements

never reaching below 73% across all days and times.

Five control measurements are conducted to verify the ability to detect changes in the context. The measurements are compared to the five-day benchmark from previous days based on the base network at Baylor University. All measurements are taken at 12:00 to allow an exact match with the benchmark, which should give the highest similarity. The first control measurement is taken in a completely different environment to validate the capability of detecting an environment that significantly changes on day 6. This measurement is taken at a grocery store, and a match with single devices of only 3% is achieved. Another measurement is retaken in an entirely distinct environment but with some devices from the base network regularly appearing there. The place that is chosen is an apartment complex with a considerable number of Baylor students. However, there are zero matches, most likely because the network is segmented into smaller subnetworks that I could not scan. Two other measurements are taken in a partially similar environment where many common devices can be expected. The chosen places are locations within Baylor University but outside of the base network, with many devices flowing between these networks. They provide a match of 5% and 0%, confirming that places with high fluctuation in the same devices are not matched. For the last control measurement, I choose our base network but during the weekend to verify that I can also detect a change in the main network context. There was less than one-fifth of the usual number of devices during the analysis, and the match was only 29%. All of the control measurements obtained values significantly lower than the threshold of 70% set in the previous paragraph. An overview of the results is presented in Table 4, including the benchmark size and the number of recurring devices found for every day of the measurements.

We evaluated the performance of this method in a network. ARP scans are used to determine the devices available. Therefore, with our method, every device receives an ARP request. I evaluate the performance in a network with 254

Simulation	Simulation match	Original match	Threshold	Simulation classification
Failure same day	78.04%	80.48%	70%	True positive
Failure day before	80.00%	80.48%	70%	True positive
Adverse device	80.95%	80.48%	70%	True positive
Attack with 15 devs.	28.95%	49.06%	70%	True negative
Spoof attack	28.95%	31.57%	70%	True negative

Table 4.5: Simulation with threshold 70%

addresses. With six devices scanning, the network simultaneously increases the network's latency (measured between two other devices) from 2 ms to between 13 and 20 ms. A full scan of the network with 254 addresses takes slightly under 3 seconds.

This verification shows that I can detect anomalies in the network and provides data that illustrate this ability in a network with hundreds of users active at the same time. It demonstrates how the 5-day benchmark was chosen as the ideal benchmark age, it explains when measurements taken at random times in an interval are better for analyzing networks than measurements taken at the same fixed times, and it describes the process for determining the optimal threshold value for this particular scenario. The control measurements demonstrate the ability to detect an unfamiliar context in numerous networks with different characteristics or at different times in the base network. This method alone cannot be used for device authentication, but it can serve as an additional factor during the authentication process. With an unfamiliar or suspicious network context, actions such as further authentication or time or resource-intensive network analysis can be taken. An example of a suspicious network is one involving the sudden appearance of a significant number of unknown devices.

4.2.2 Simulation

In this section, I simulate the network's behavior in potential situations that did not occur during our five-week real-world evaluation but are of significant concern. For the simulation, I use the real-world network measurements, and I adjust them to the particular scenarios by removing or adding the devices into the measured data. I explore cases that could potentially lead both to false negative and false positive classification. For initial simulations, I choose day 11, time 12:00, from our measurements. For latter scenarios that could lead to false positives, I choose the

Saturday following day 10 and again 12:00 time as I have data for it in the control measurements. Results are summarized in the Table 4.5 and described bellow.

The first simulated case is a failure of the stable device. This can be divided into two events. The device can either fail before the measurement is taken, which means that it is not included in the current benchmark. Alternatively, it can fail on the same day and therefore is included in the benchmark. Failure on the same day decreases the number of recurring devices from 33 to 32, and therefore match decreases from 80.48% to 78.04%, which is well above the threshold. Failure of the device in the preceding days decreases both benchmark size from 41 to 40 and number of recurring devices to 32, which leads to 80.00% match. Again above the threshold, I set. The only measurement where failure on the same day would lead to a false negative is day 21 in 16:00 as it would decrease to match to 68.23% (failure on the day before would only decrease the match to 72.00%).

The second scenario is when an adversary is present on the network from the beginning. This leads to the increase of the benchmark and stable devices found. In our simulation, it increases match from 80.48% to 80.95% with benchmark increase of one to 42 and number of recurring devices increase to 34.

The third case simulates a broader attack on the network, with malicious 15 devices present on the network. This increases the number of devices on the network to 133, the benchmark size to 53, and the number of recurring devices from 11 to 26. It leads to a match of 49.06%, while the match without the attack was 28.95%. Given our network and the specific day, an attack would need to consist of 52 devices to reach our threshold and thus lead to a false positive.

The fourth case simulates an attack where the malicious devices spoof the MAC address to one of the benchmark addresses not present on the network. The device's presence increases the number of recurring devices to 12 and the match from 28.95% to 31.57%. Eleven devices in a coordinated attack would be needed to lead to false positive. Therefore I identify this as the weakest part of our method, as 11 devices are considerably smaller than 53 devices from the previous scenario. Also, those 11 devices can be present on the network only during the attack, and therefore they will more likely stay unnoticed by network administrators.

4.3 Threats to validity

Experimental verification presented in this study is based on an experiment with one selected network and a simulation of various situations that can occur during network operation. This can be considered as a threat to validity. Although the network used in the experiment was sufficiently extensive, it cannot be assumed that other large networks will have a similar topology and characteristics.

However, this issue can be mitigated by adjustment of parameters of the proposed methods. In networks where devices do not fluctuate as much as they do in university networks or in networks where there are many newcomers or irregularities, the values for the threshold, the optimal benchmark size, or the measurement times may vary significantly.

Another concern may be raised regarding the fact that I used MAC addresses as a device identifier in the proposed method and the experiments. Generally, MAC addresses are easy to spoof, and if attackers determine the set of recurring devices, they can spoof them in the network, which would lead to a false positive result.

To mitigate this issue, alternative device identification can be used. With an alternative identification of a device, the principle of the method does not change.

4.4 Discussion

The *threshold* given by the Algorithm 1 can be further adjusted to modify the behavior of the method. Lowering the threshold decreases the number of false positives, increasing the number of false negatives. Increasing the threshold has the opposite effect. Each percent I remove from the threshold determines the percentage of devices that are allowed to fail without a false negative. For instance, this could provide a safety margin while decreasing the accuracy of the method.

The *number of benchmark* days determine the adaptability to network changes. Networks with a higher number of fluctuating devices will have a lower value than networks where the same devices are present all the time. Those values can be modified to suit the particular network.

Timeframes definition affects the behavior characteristics of the method. Basically, the longer the timeframe, the more devices fluctuate. While this can offer some extra protection against MAC spoofing, it decreases the threshold and, therefore, can lead to false positives.

The proposed method is dependant on the size of the network. At least tens of overall devices are needed to provide meaningful results and hundreds to achieve a consistent output.

The described approach provides an *additional authentication factor*, and therefore, it would not be sufficient as a standalone authentication method. Also, the method does not detect changes in the behavior of the devices themselves but in its network neighborhood. Therefore, the proposed method does not detect device hijacking.

4.5 Summary

The proposed solution allows determining context for all types of the IoT devices. The case study proves its feasibility and usability. It makes decisions based on changes in the context in the network around devices, and therefore, it can detect suspicious or even malicious behavior. It is a simple mechanism in terms of device resources, and it can be deployed on every IoT device capable of communication over TCP/IP, allowing system operators to inspect the network and, if needed, to take appropriate actions to resolve an issue. The context information can be used as an additional security factor in conjunction with existing security architectures.

Performed real-world experiments demonstrate the feasibility of the approach in a network with a significant number of devices. The results indicate that the concept can provide valid results and increase the security of both the devices and the entire network. This sort of approach especially fits for secure locations, such as laboratories, energy sources, or military bases, where the aim is to limit external devices. However, this method might not be the best for locations where devices have a high churn rate, such as shopping centers.

Chapter 5

Context-aware authorization

While having contextual information is a crucial prerequisite of context-aware security, the sole fact of having access to context does not make the security of the application context-aware. Currently, the most prevalent authorization architecture RBAC does not support context-awareness as it is a pure abstraction in the form of roles over permission assignment to the users. A similar situation is with MAC and DAC where only permissions are assigned to the user without any contextual conditions.

Application owners and operators, as well as software developers, are well aware of the added value of the context-aware authorization. Nevertheless, even there are numerous proposals for context-aware authorization, none of them is widely used [56], [59], [60], [62]. They are not used more frequently because they are either too complicated for practical use or too innovative, requiring the whole authorization system redesign, which is challenging to incorporate into an existing solution, both from engineering and security auditing perspectives.

In this chapter, I describe my research on extending RBAC with context-aware elements. The extension bases on users' security levels, which are quantifying the user's context. To access resources require the user to possess a particular level in addition to her usual access rights. This proposal allows an extension to existing RBAC solution and architectures with context-aware elements.

The achievements of the research can be summarized into those points:

- Extension of traditional RBAC with context-aware element
- Implementation of the proposal into open-source Identity Management (IDM) and security management solution
- Demonstration of the approach on the use case and comparison with traditional approach.

The research represented by this chapter has been published in two conferences. The initial idea in the submission[A.10] to RACS'15 and its extended version[A.9] was presented on SAC'16. The later paper is has seen a good impact on the scientific community as it is fairly well cited - 23 citations, and seven out of them are from articles in impacted journals.

5.1 Proposed Solution

Authorization policies in organizations tend to be very consistent and are changing just slightly over time, if at all. Most of the organizations do not want or do not even need to apply any radical changes. Therefore, context-aware authorization must be another logical step to evolve current security. This will allow us to build new authorization rules on the existing and well-proven solutions, and it also makes the solution more accessible for people who are familiar with current solutions.

I propose the creation of a security level, which is based on context. This serves as an addition to traditional roles in RBAC. The level can be understood as quantification of how the user is trustworthy, and it is dynamically tied to the user and his context. The security level creates a second authorization constraint besides traditional security permission. Therefore resources in an application can subsequently have two different kinds of authorization rules - classic policies tied with roles and a security level. Both of the approaches are independent and complement to each other. Having one without the other is possible, though using only context security without other security policies can be unpredictable and therefore delicate to define in a production-grade ready application.

As the user's context and the application changes, the level needs to reflect the dynamic nature of the context. There are several moments when the level calculation is possible. The first moment is to calculate the level during the user's account creation. However, this does not reflect the dynamic nature of context and therefore is unsuitable for our needs. The opposite extreme is to determine the level on every authorization request. This would reflect changing context most reliably, but it is very demanding for computational resources and also time-consuming, as the context check might not be trivial. The best compromise seems to be to determine the level during the user's login into the application. Figure 5.1 shows a system sequence diagram of determining user level and its storage for further use. It decreases the number of context checks by several orders, and at the same time, it provides a very accurate snapshot of the user's context. In cases when the context changes rapidly, the user can perform relogin, or even the application can

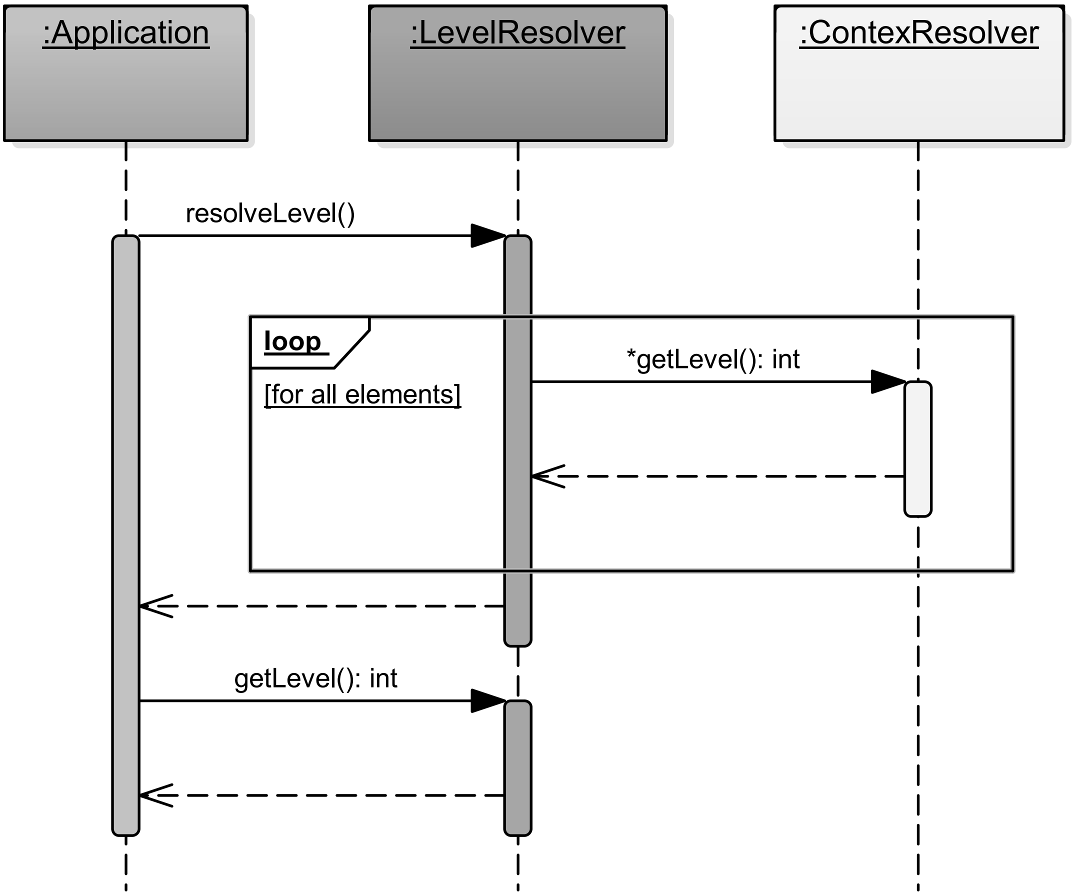

Figure 5.1: Process of determining security level

enforce a new level calculation manually.

Context resolvers achieve the level resolution as shown in Figure 5.2. Each resolver takes responsibility for checking one particular part of the context. For example, one resolver would determine the network context from which the user comes. Another would check the time of the day and so on. Every resolver would return the level it grants to the user. As the security resolver is written within the application, it has access to the user's information (e.g., his request, information about him stored in a database), as well as it can use information about the application (e.g., number of requests, number of users).

Furthermore, it could even consider the machine the application is running on (e.g., a load of the machine, resource usage, location of the server). The final level is not set in the resolver, and it does not decide just if to grant it or not; the resolver itself makes the decision, which level to grant based on its own knowledge and logic. After every resolver performs its inner logic and determines the level on

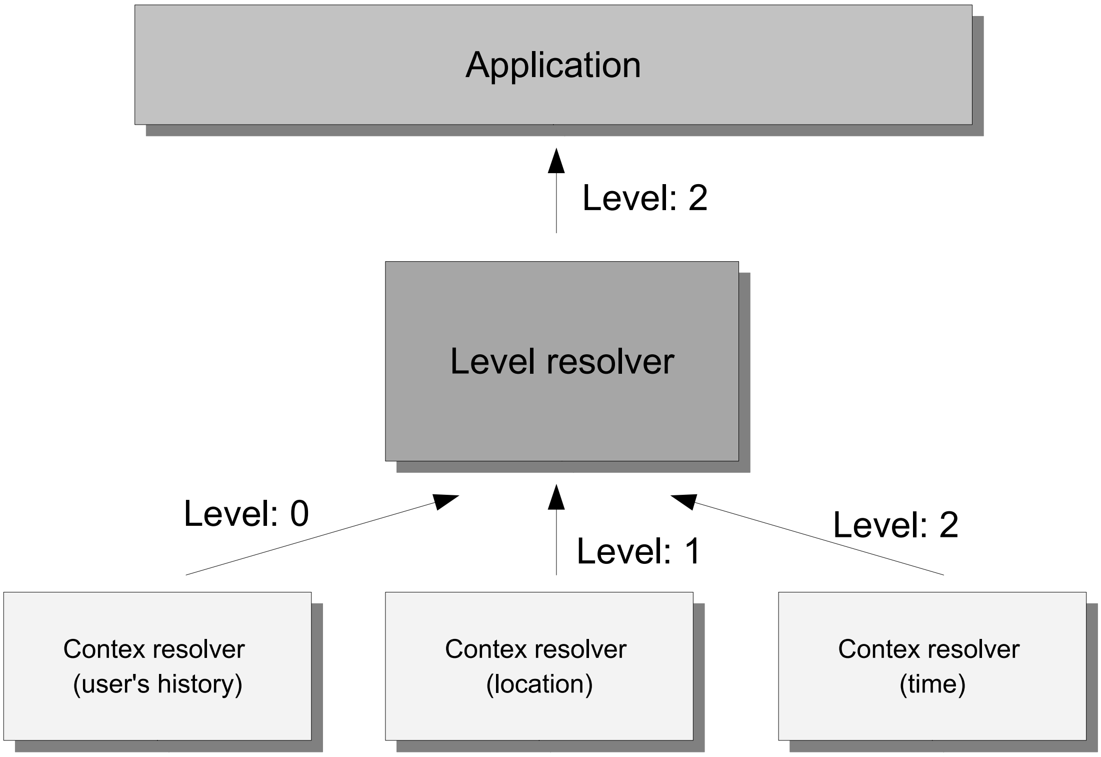

Figure 5.2: Level determination from given context resolvers

its own, the highest level is used as the final user's security level.

```
@AllowedRoles('admin','manager')
@RequiresLevel(3)
public Resource getResource(int Id) { ... }
```

Listing 5.1: Sample use of security levels for resources

The level representation by itself is very abstract. It is only necessary for the level to be comparable with other levels to determine whether the given level is higher or lower than the required one and determine the highest one. Therefore it is not crucial whether number, string or even some more complex structure represents the level. This leaves much space for customization for a given application.

Listing 5.1 shows usage of the levels in the code. You can see the definition of allowed roles to access the method as is common in RBAC. In addition to this, you can see the required security level that the user needs to possess to be able to invoke the method.

The proposed solution has many advantages. The most important ones are:

- Lightweight - it does not require any complex structures in the application, nor does it not consume significant system resources.
- Easy to use - it just requires adding another type of constrain to resources that need to poses context-aware authorization.
- Voluntary - if someone wants to use plain RBAC he can, and just to chosen resources, he might add level restrictions.
- Scalable - there is no predefined set of levels, nor is there a limit in the number of application levels.
- Universal - the solution can be modified and used with other authorization architectures, not just with RBAC.

```java
public class NetworkContextResolver implements ContextResolver{
  ...

  public int resolveLevel(){
      int match = this.networkContextService.getMatchByDays(3);
      if(match) >= 50){
          return 2;
      }
      return 0;
  }
}
```

Listing 5.2: Sample use of security levels for resources

However, the solution poses few limitations, which need to be worked further on. Among them, the most significant are:

- Hard to determine exact context - sometimes it can happen that some resource should be accessible just from a given context. For example, some resources are accessible only during the day and some just during the night. Such scenario is impossible to secure with the proposed solution.
- Levels are linear - the structure of the levels is strictly linear, and therefore it is impossible to build some tree or even more complex structure of levels. Often happen that there are multiple context rules, which are granted a different set of rights. Levels cannot model, for example, a geographical situation when users from the same state have some rights, but people in different locations of the state got additional specialized rights.

User's status	Actions	Obtained
none	Browse e-shop	default
logged in	View order history	username/pwd
verified	Pay for purchase Change delivery adddress Set trusted IP	SMS code verification Access from set IP

Table 5.1: User's status and allowed actions

Level resolver can be used together with the network context described in the chapter 4. Listing 5.2 shows an example of a resolver that uses `NetworkContextService` to determine a perceptual match for the given number of previous days. If the match is above 50%, it returns level 2, if not, zero is returned.

5.2 Experimental verification

The solution described above was implemented into the open-source project PicketLink [165], and it is part of its released codebase. PicketLink was an identity management and security framework focused on compatibility with Java EE specifications. During my doctoral studies, the project has merged with KeyCloak project [166].

To demonstrate the value of my approach, I create two prototypes of a simple e-shop: the first using the proposed solution, and the second one relying on traditional security methods. Then I compare the implementations and point the differences and increased effectiveness on my proposal. Both variations of the application are developed using Java EE 7 [167] specification.

Both approaches' security functionality is the same from the user or administrator perspective and contains multiple actions and different authorization rules. Users without any form of authentication can browse the items in this shop and add them to a cart. Users who have logged in using their login name and password can view their order history and delivery address. Finally, there is a third level of authentication of the user called "verified user". This status allows user to change their delivery address and to pay for the purchases. This security level can be obtained by additional authentication done in one of two ways. The first possibility is to use a specially generated verification code delivered to the phone by text message. A second possibility is that the system allows a user to set a trusted IP address (it can be set only if the user is already verified). When the user logs in from that IP address, he/she is automatically considered verified.

The application is very simplified and contains only few actions (represented by secured service layer methods). Table Table 5.1 summarizes them for every user status and also shows how the security status is obtained. It is evident that the authorization rights are simple for this application; however, they most likely will be very complicated for real applications.

```
@HasRole('customer')
public void makeOrder(Order o) throws NotTrustedUserException {
  if(!ipCheck.isIpTrusted()&&!smsCheck.isSmsVerified()){
    throw new NotTrustedUserException();
  }
  ...
}
```

Listing 5.3: Method secured traditional way

In implementation without levels, every secured method needs code for determining user's context. As Listing 5.3 shows it brings few lines of unrelated code into those methods as well as new declaration of thrown exception. Code exhibits obvious concern tangling [168] represented by classes `IpCheck` and `SmsCheck`.

```
@HasRole('customer')
@RequiresLevel('2')
public void makeOrder(Order o){
  ...
}
```

Listing 5.4: Method secured with levels

Implementing the same logic using proposed levels is displayed in Listing 5.4. It is clear that the method using security levels is significantly shorter and does not have any unrelated code inside. Concern separation [168] increases cohesion [169] of method and at the same time reduces coupling [169]. The class `IpCheck` has been changed to a level resolver, which reduces dependencies as all the resolvers are invoked automatically during login. The class `SmsCheck` was deleted completely because the framework allows setting up the level in authenticator as is shown in Listing 5.5. The Listing 5.4 demonstrates that the approach with levels adds to code of secured methods just a single line with annotation. Besides, it keeps the code for determining level separated from the application's business logic in a separate package. All of this contributes to faster development once the levels are set up as well as easier maintenance and testing of the code. Without using levels, there needs to be a condition for every contextual check inside the given method. Therefore, the complexity of the code is unnecessarily increased, and readability decreased. Even if the authorization rules were extracted to another class, it would add one more dependence for the given class. The proposed solution can also decrease the number of total classes in application because some levels are determined automatically by annotations (e.g., over authenticators).

```
@SecurityLevel("2")
public class SmsAuthenticator extends
BaseAuthenticator {
  ...
}
```

Listing 5.5: Authenticator for SMS verification

In the given example, the implementation with levels removes three code lines and exception declaration while adding one annotation in half of the secured methods. It also deletes one class (while adding one annotation to the authenticator). The second class is changed, and there are no dependencies to it. It is very likely that with more complicated applications, the benefits will be even more significant. The case study result can be summarized as follows: better reuse, lower coupling, higher cohesion, less code (about three lines of code per rule usage and about 10 per rule declaration). Code savings can be significant in large projects. For example, a project with 100 authorization rules, each used 300 times, saves almost 2000 lines of code.

5.3 Threats to validity

The research results are validated only in a single case study with limited size. Though the results are part of an open-source library, it is unclear whether it saw a production issue.

Having the levels linear can be a limitation for its production usage. Over the time, since this research was published, other prospective methods for context-aware security appeared. As an example, I can name ABAC that might provide similar outcomes with more flexibility.

5.4 Summary

The research presents a convenient way to enhance RBAC architecture with the context-aware element. The context-aware architecture aspect is represented by a security level, which is a linear abstraction of trust based on the user's context. To access a resource in the application, the user must possess not only the required role but also the required (or higher) security level. This solution keeps the advantage of the RBAC architecture while enhancing it with context awareness. Though no research has been made to support this hypothesis, I believe that the approach is easily portable to various other security architectures.

This research's theoretical results led to the open-source contribution that both validated our approach from an engineering perspective and enabled us to implement case study faster. The case study demonstrates that our approach is feasible and brings significant and apparent benefits compared to plain RBAC with manually added contextual functionality.

Chapter 6

Security rules sharing

The IoT is built on an idea that multiple devices are cooperating together to reach a common goal. The devices in the IoT network may alone be cheap, single focused and expandable, but when coordinating together with other devices, the whole ecosystem's value dramatically increases. Therefore, an individual device needs to trust other devices to safely communicate with them, trust their information, and ultimately to deliver value for a user. The trust must be established not only among devices and but also between the user and the devices.

However, device management and creation of a confidential environment between them is one of the major open issues in IoT [170]. IoT can be divided into three layers - perception, transportation, and application. The device identity management must be implemented at least on the application layer. The layer is responsible for all communication with the end user and a significant part of communication with devices, as it gathers all relevant data for the user. Though, implementing it on other levels too can gain additional benefits.

This chapter presents my research conducted on this topic. I propose a framework for device authentication and essential identity management. It consists of a centralized identity store, and it is using already existing security standards and technologies. The centralized solution allows response fast enough to prevent any further damages in case of an attack targeting devices [171], while reusing existing technologies allows smoother and faster adoption.

To illustrate the need for such a solution, let us imagine the following situations with a smart car. Initially, there is a car equipped with a location sensor and connection to the Internet. Such a vehicle could provide its location on request. In the base configuration, the location could be used only in emergencies. However, later we may want to change the settings. E.g., the car operator decides to participate in any form of smart transportation. Alternatively, an insurance company offers an owner a lower rate based on his small annual mileage. Having a central identity store, in this case, would make everything easier. Car operator would allow devices with the role “insurance locator” or “London smart traffic” to communicate with the vehicle.

The main accomplishments of my research are:

- Describing centralized IoT authentication and IDM system
- Implementing the proposal for case study and verifying the results
- Evaluation performance overhead of the proposal

The research described in this chapter has been initially published as a conference paper [A.5] and later extended into a journal article [A.4]. Both papers received decent recognition from the scientific community as they are cited, including publications in journals with impact factor.

There exist other journal articles [85], [100], [106] by authors that I have never had any interaction with. Their papers are presenting very similar results. Their research has been published at least a few months later after my conference paper, which suggests that we came to the same results independently. Getting multiple identical results from multiple separated research efforts validates the results and can be used to prove the validity of the results.

6.1 Proposed solution

The research led to a central identity store solution, which would keep a record for every device connected to the network. The central element contains unique identifiers for devices and their credentials, but it also supports the RBAC by storing the roles internally. All machines and applications in the network can use those roles for their authorization rules. The trusted central identity provider creates an environment in which both participants can verify the other partner’s identity, and they can also determine if the partner is allowed to perform the given action.

Using the central identity element in IoT promotes a trusted environment. Devices do not deal with a machine to machine trust; it is enough to establish confidence in the identity store. Whenever there is a suspicion about a hostile takeover of any device, the device can be disabled with a single action. This action ensures immediate propagation through the whole network. This kind of approach also applies to less severe situations, such as device malfunction resulting in transmitting incorrect data. However, using any central element in network architecture has

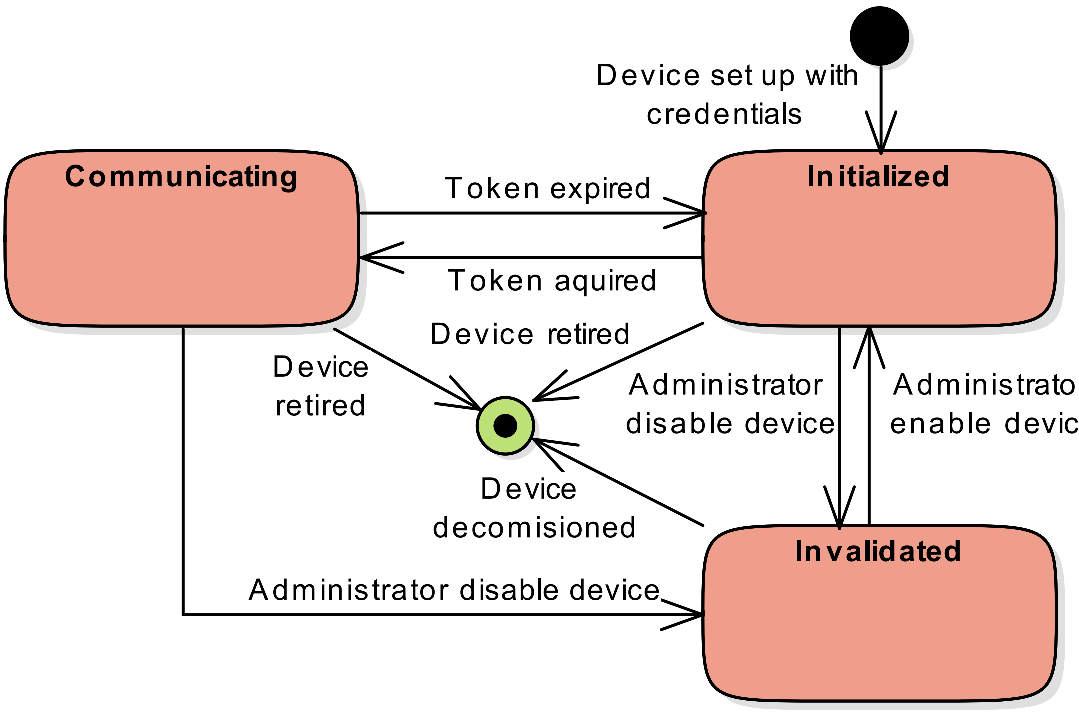
Device set up with credentials
Communicating
Token expired
Initialized
Token aquired
Device retired
Device retired
Administrator disable device
Administrator enable device
Device decomisioned
Invalidated
Administrator disable device

Figure 6.1: Diagram of device's possible states

known security threats, for example, Denial-of-Service attack, and therefore need to be sufficiently protected.

Communication (not only) in IoT consists of two participants. The first one, called provider, exposes services to others. In the proposed method, the provider must register at the identity store as an identity client if it decides that its services are confidential. The second one, the consumer, uses the provider's functionality and initiates the communication.

The consumer needs to have a registered identity in the central identity store. To initiate communication with secured service, the consumer authenticates using an identity store and retrieves a token representing his identity (and possibly other information, as roles), signed by the identity store. Later, the consumer uses the token for communication with the service provider, which validates the token using the provided signature. This enables authentication of the consumer with a trusted element, and therefore, it prevents misuse by malicious service providers.

The identity store does not need to serve solely as an authentication service; it may provide additional functionality. For example, it can provide additional data used for authorization. I focus on roles for RBAC, but generally, any information is possible to be provided by the central store, e.g., attributes for ABAC. Then the

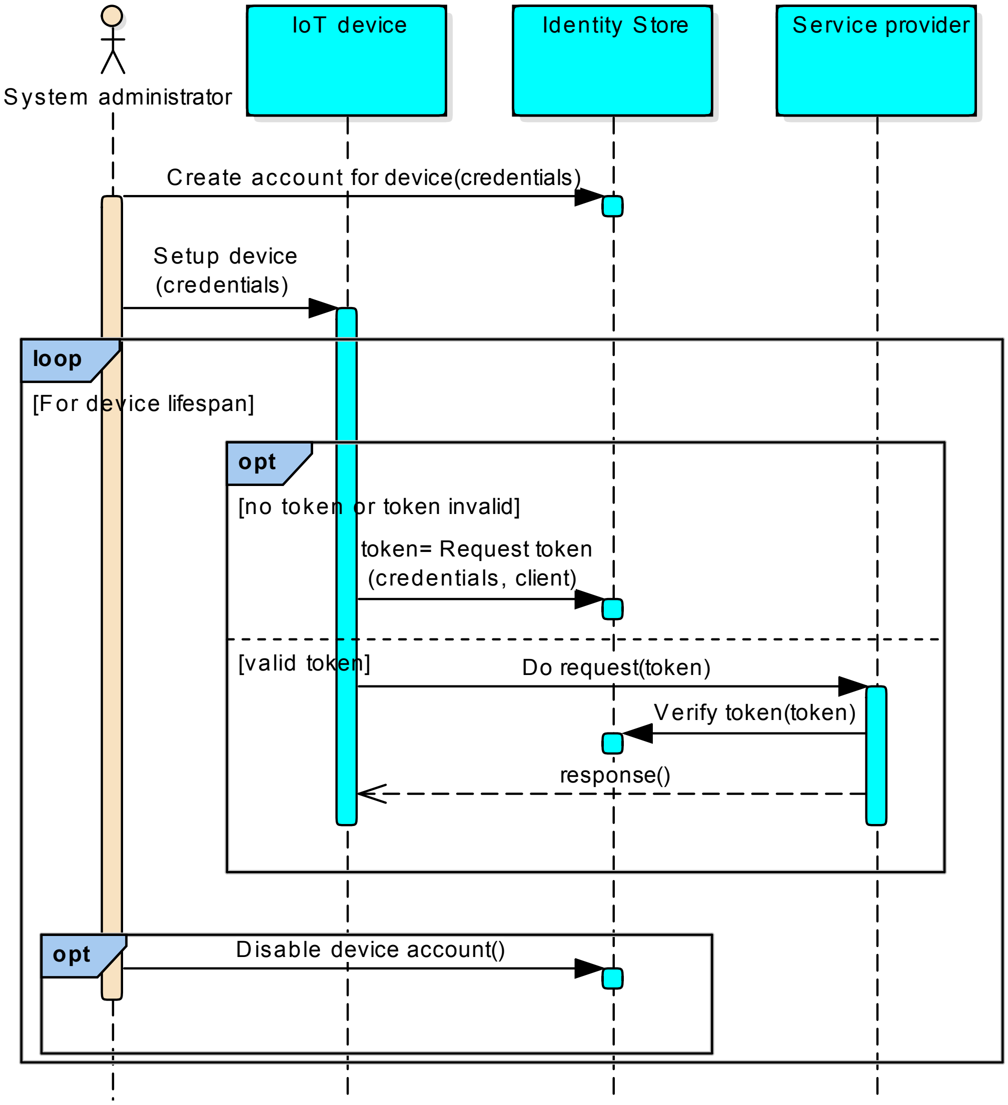

Figure 6.2: Diagram of communication in the proposed solution

service provider can specify roles required for a given action. When the consumer tries to use the service, its roles are verified with the identity store. This also means that the roles are global for the specific IoT environment, which reduces efforts related to administration and the number of repeating configurations across all systems.

Additional information about the device is stored in the token returned by the central store. The token is signed, and receiving application verifies the token using the central identity. This is especially useful when communicating with stateless services, as the request contains all required information for authentication and authorization about the service caller.

Figure 6.2 demonstrates the workflow devices authentication and authorization of devices. The following steps describe it:

- Administrator creates an account for a device and set up its roles.
- The device is configured with credentials provided by the administrator and requests a token from the store.
- For any confidential communication, the device uses the token to authenticate itself.
- Application/device receiving the communication verifies the identity and roles by given token at the central store.
- Administrator can disable or remove a device from the identity store and therefore effectively disable it for any cooperation.

Configuring a device in such a network does not require significant effort. First, the device is setup provided with credentials. Before it initiates communication with its partner, it requests a token with credentials, valid solely for the given service provider, restricted to a certain period of time. In some instances, a time-unlimited token is viable; in others token with a short-time validity is preferred. However, once the device obtains a token, it can communicate freely with the partner. The partner can verify device identity as well as its roles, based on the presented token.

The solution itself is composed of two parts: administration application consisting of user interface and IDM server itself and then the library for IoT devices consisting mainly of the communication module. The communication between the modules is done via the Internet over the family of HTTP Internet protocols [172]. However, support of additional protocols, like MQTT [173] can be added easily. The communication inside the modules is done through native Application Programming Interface (API) of the given programming language. Figure 6.3 shows a component diagram of the suggested architecture.

- IDM module – a module that administers devices and their roles. It also verifies tokens.
- Device provider – allows devices to log in and refresh token
- Administration provider – a module that enables an administrator to add, remove or disable devices

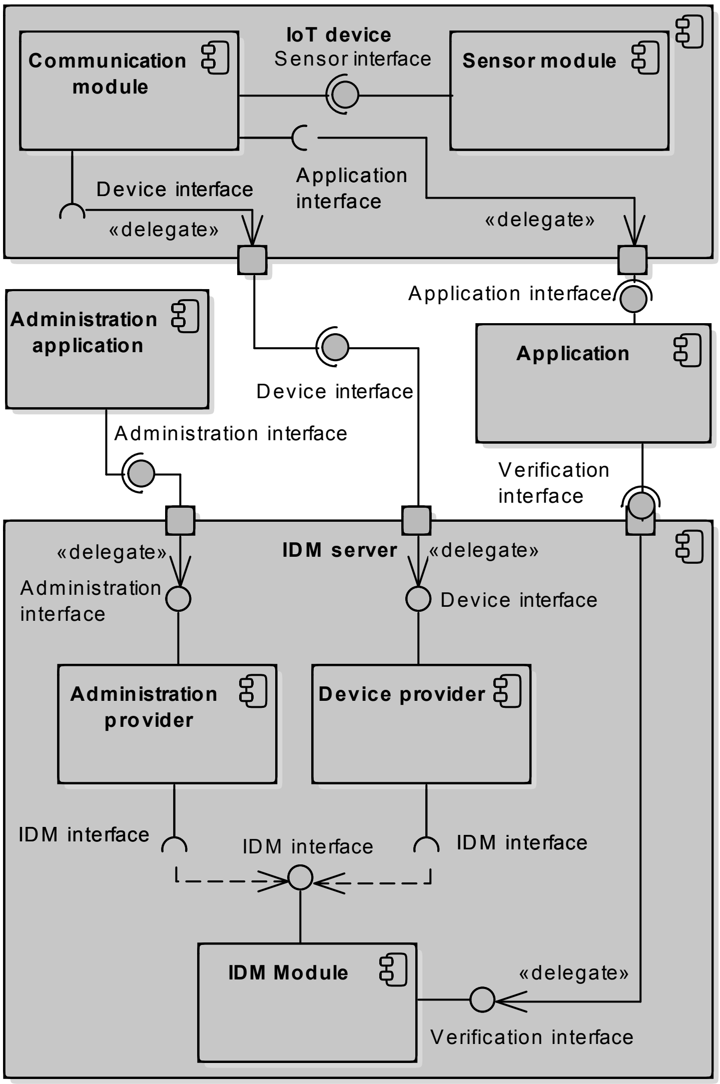

Figure 6.3: Component schema of the proposal

- Administration application – an application that provides User Interface (UI) for the administrator
- Communication module – module which is embedded into the IoT device and takes care about every communication. It authenticates the device, retrieves the token, and uses it for further communication.
- Sensor module – This module contains the business logic of the IoT device.
- Application – Application that that uses data from the sensor. It needs to verify the device's token against IDM server.

The framework supports communication over REST API [142]. This allows utilization of all the technologies and properties of the HTTP protocol [172]. At first, SSL protocol [174] is tightly integrated with the HTTP protocol (called HTTPs [172]), which provides us transportation security as added identity confidentiality in the network. The advantage brought by this approach is that firewalls rarely block communication on ports 80/443. A potentially more suitable protocol might exist than HTTP(s), which is tied to REST architecture. However, none of them is so widely used and adapted as HTTP(S).

6.2 Case study

Based on the framework proposal described in the previous section, I have created a prototype that builds on existing solutions (as suggested by Finkelstein [175]) integrated together to provide the expected functionality. Building on top of existing solutions allowed me to leverage existing experience and simplify the transition to possible real usage. Furthermore, using existing infrastructure allowed me to focus on the novel approaches than re-engineering already solved challenges, and mainly it brings a verification to the proposal's applicability and integrability with existing production-level tools. Moreover, it ensures that the current state of the art is sufficient for an extension, and no other crucial technologies need to be developed as a replacement.

Roman [170] states that also traditional Web 2.0 Single sing-on (SSO) such as OpenID [46] or Shibboleth [176] could also be used in this situation, although it should be noted that they were not designed to fulfill certain IoT requirements such as identity disclosure. Therefore, I have opted to try out existing technologies to determine if (and how) they are sufficient for usage in the IoT ecosystem.

Small scale simulation of IoT was created for the purpose of that paper. Figure 6.4 shows a scheme of our case study application. It consists of those major elements:

- Central identity store – Keycloak [166] was chosen as it providesSSO and IDM for web applications and mainly for RestFul web services. For the purpose of this case study I have leveraged mainly support of Oauth 2 [150] and OpenID Connect [46] JWT [151] standards.

- Two sensors – specifically movement sensor HC-SR501 and temperature sensor DS18B20 were used. Both of the sensors provide digital output and, therefore, can be used without any analog-to-digital converter. However, sensors still need to be connected to some device with computational capabilities to transmit

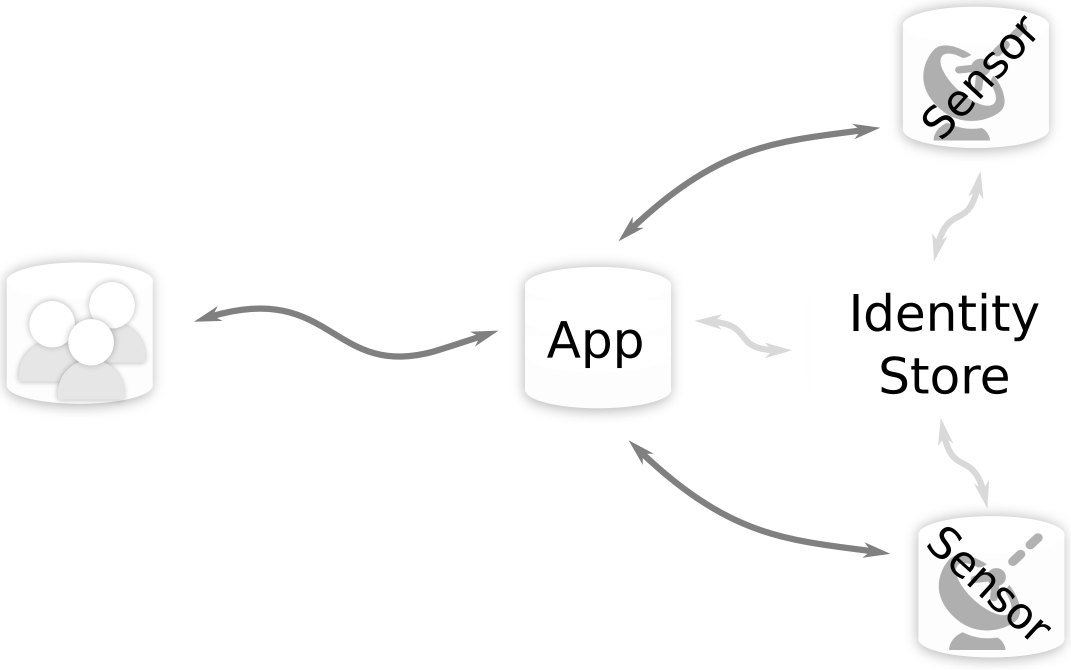

Figure 6.4: Scheme of our case study application

the data over the Internet. In this case, Raspberry Pi is used to host sensors' services.

- An application using data from sensors – simple application with RESTful interface. It gathers data from sensors and exposes them to users via JavaScript web front end.

Central identity store is deployed as a standalone application. It contains two roles `temperatureSensor` and `movementSensor`. Next, an account for every device was created and assigned appropriate roles. The password and username of the given account must be provided to the particular sensor. Authentication token expiration needs to be handled, but I have chosen to never let it expire for the sake of the simplicity of the case study. This allows a device to use it as long as needed without the need to refresh it periodically. OAuth 2 protocol is used for bearer token [177] acquisition, and the token issued follows JWT standard. The advantage of using JWT tokens is that they contain additional information, such as user roles. Therefore, the communication with the central identity store can be reduced to a single call aggregating multiple information and thus improving the performance.

The sensors by themselves do not possess any computational power, and therefore they need a device to control and observe them. In this case study, they are directly wired onto the bus of Raspberry Pi computer. A small script written in JavaScript on top of the Node.js framework performs all the sensors' logic. The script differs for various types of sensors and needs to be initialized with credentials for the particular sensor. The communication process is following:

- Acquire token with username/password after start-up.
- Every second sends information to the central application, with the token for authentication and authorization.
- If token becomes invalid, attempt to re-authenticate.

All the communication between sensors, the central identity store, and the central application are made through RESTful interfaces. As the sensors are managed by JavaScript service, additional mocked sensors were deployed into the case study environment, not impacting the infrastructure's scalability.

The central application receives data from sensors and displays them on a web page. In order to do so, the application consists of two parts - backend and frontend. The backend part uses Java EE, and it leverages the Keycloak adapter to make integration with the central identity server easier. It provides a RESTful interface for gathering data from sensors and also for exposing the gathered information. The frontend part of the application is also connected to this RESTful interface.

The case study demonstrates that using the proposed scheme is possible, and it enables the expected advantages, such as broad machine to machine trust and rapid incident reaction. However, it also shows limitations that should be addressed. First, there is a need to distribute credentials for every sensor, store it at the device and use it for obtaining a token. Second, an administrator needs to manually create an account for every sensor, set up its roles, and propagate identification and password to the sensor.

6.2.1 Performance evaluation

The performance overhead of our case study is very low. Our measurement shows that it takes from 115ms to 130ms (with a mean time of 123ms) to retrieve or refresh the token. This was measured in the Node.js program controlling the sensor. Validity of the token can be determined by the system administrator and can very from a single request up to unlimited. An illustration of the amount of that device spends managing security token can be seen in the Table 6.1. As I can see, the overhead would become significant only if the data were sent from the sensor every second with a single usage token.

Token validity.	Percentage of device time in the worst scenario
1 second	13 %
1 minute	0.21667%
5 minutes	0.04333%
1 hour	0.00361%
1 day	0.00015%

Table 6.1: Overhead of sensor communication.

Another perspective worthy of consideration is network usage. The volume of the data sent exhibits, at first sight, a significant increase. The data are transferred from the sensor using HTTP GET method. It does not contain anybody, and therefore the length of the request is small. For example, the temperature sensor's requests are only 189 bytes large. Out of the total, there are 152 bytes for URL address, 6 bytes for the data itself, and 31 for various symbols needed in the HTTP request, such as headers. With security added, the HTTP request changes the size to 1398 bytes. The token by itself is 1185 bytes long. It may look like a colossal overhead; however, 1kB of data added is an insignificant increase with current Internet technologies.

6.3 Threats to validity

The measurement for the case study was performed in a small environment. The application ran on the same computer as was used for the user's connection and validation. The sensor network consisted of 2 real sensors - one motion sensor and one temperature meter. Both of them were connected to Raspberry Pi that administered both of them. I did not have sufficient resources to simulate a large scale IoT environment. In such a case, the performance is questionable. Based on the performance of the Keycloak, I firmly believe that thousands of sensors should be manageable. However, I cannot claim where the limits of the solution are, whether it is in order of tens of thousands of sensors, hundreds of thousands, or even millions.

Based on the same issue with the small testing network, I did not try more than two roles for authorization. There is no doubt that devices and central stores can manage significantly more than any device would ever need. Nevertheless, there is still a need to administer them. I used RBAC system, which can become hard to maintain with an increasing number of roles. I assume that more than 100 roles would become hard to manage. However, there is currently no research stating how many roles would be needed for IoT environment.

6.4 Summary

The suggested solution addresses IoT device management with the main focus on the centralized authentication and partially focused on centralized policy definition point. This proposal is built around a centralized OAuth 2 [150] server that administers all the devices and allows for the definition of their roles. The chosen authentication protocol allows then all the devices in the network to securely communicate in the environment. Centralized nature of the authentication enables fast reaction in case of any adventitious event.

I have implemented the solution using Keycloak [166] and few device clients to prove its correctness. The results indicate that the approach is feasible, reasonably simple to implement, and mainly does not bring considerable overhead.

Chapter 7

Conclusion

Conventional security architectures and approaches are unsuitable for IoT security for multiple reasons. They do not scale well, they are not prepared for a heterogeneous environment, and they were not built with constrained devices in mind. Also, they do not leverage IoT advantages as broad access to the context. With the increasing popularity and prevalence of IoT solution in past years, the issue of IoT security became prominent.

During my research, I have focused on extending traditional security approaches with context-aware elements, transferring them into IoT environment. I have also proposed a method for context retrieval for IoT devices. Overall, I have developed a solution containing context resolving, adapted existing RBAC security architecture to consider contextual information, provided a method to share and propagate new or updated security rules across the IoT devices without additional overhead, and I have participated on testing the IoT solutions.

The specific contributions of my Ph.D. research can be summarized as follows:

1. Survey with a broad overview of the existing security research in the IoT domain. The survey not only lists the most recent IoT research but also categorizes the research into multiple categories, provides an overview of what research has the most impact, and analyzes trends.

2. Method of determining a context of IoT devices from its network neighborhood. Devices use a snapshot of the network state containing all the available devices. This snapshot is then examined and compared, and significant deviation from the normal state is used for subsequent authentication (or authorization) rules. This method is largely customizable with various parameters, and it is applicable to any device communicating over the Internet network.

3. Enhancement of (mainly) RBAC with elements of context-awareness. I added

another dimension to the architecture that describes the context. The context is expressed through a single-dimensional property called "security level". The results of this effort were part of the open-source security and identity management project PicketLink [165]. With minor changes, the solution would work with other security architectures. This research [A.9] also got the biggest impact on the scientific community, as the results are presented in the article that is most cited from all my articles.

4. System for sharing authentication and authorization rules in the IoT environment. This system uses existing solutions, namely OAuth 2 [150], OpenID Connect [46] and JWT [151] to propagate security rules. The rules are stored in a centralized authority that acts as a single source of truth for the security policies.

All of the work was published in reputable conferences and peer-reviewed journals. All of the code I created is publicly available, either as part of the open-source project, public git repository, or as an attachment to the articles.

7.1 Future work

The results of the research conducted during my study open multiple opportunities for future work. Initially, a possible research direction is to use some form of an AI algorithm to evaluate gathered contextual information from the devices. It can be either used to determine the algorithm's parameter values, or it may even remove the need of the parameters, and the AI will evaluate the security threats.

Context retrieval I presented uses network neighborhood, which is the only subset of all existing device context. In the IoT environment, various devices can recognize other contextual through their sensors. It would be beneficial to explore the possibility of retrieving the context from other devices and correlating it with the given device (or user). This would require both mechanisms for context sharing from devices and methods for approximating this context's relevance to other network participants. I have conducted the first experiments [A.11] in cooperation with an undergraduate student, but the idea was left unfinished.

For the security architecture proposal, I have enhanced RBAC with security levels. While this allows for basic context-awareness, it can not express a more complex state of the context. It would be interesting to adapt the solution to usage with ABAC and represent the context as attributes.

Security rules sharing I described is done using existing conventional methods - OAuth 2. It would be valuable to explore methods of how to propagate the security rules without the need for a centralized element. This would require creating a method to describe the rules in a standard format, a mechanism to discover and share them in the network, including their verification to prevent malicious rules and attacks, and then developing an engine that could apply the rules.

Finally, it would be interesting to explore the possibility to describe the IoT participants using a directed graph. The graph would capture the ownership relationship, types of the devices, required protection, connections with other participants, and other relevant information. Based on this topology, we could develop a mechanism to determine correct security rules and places where to enforce them.

Bibliography

[1] F. Loi, A. Sivanathan, H. H. Gharakheili, A. Radford, and V. Sivaraman, “Systematically evaluating security and privacy for consumer iot devices”, in *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy*, ser. IoTS&P ’17, Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 1–6, ISBN: 9781450353960. DOI: `10.1145/3139937.3139938`. [Online]. Available: `https://doi.org/10.1145/3139937.3139938`.

[2] R. Anderson and T. Moore, “The economics of information security”, *Science*, vol. 314, no. 5799, pp. 610–613, 2006, ISSN: 0036-8075. DOI: `10.1126/science.1130992`. eprint: `https://science.sciencemag.org/content/314/5799/610.full.pdf`. [Online]. Available: `https://science.sciencemag.org/content/314/5799/610`.

[3] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey”, *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010, ISSN: 1389-1286. DOI: `https://doi.org/10.1016/j.comnet.2010.05.010`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S1389128610001568`.

[4] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, “Internet of things security: A survey”, *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, 2017, ISSN: 1084-8045. DOI: `https://doi.org/10.1016/j.jnca.2017.04.002`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S1084804517301455`.

[5] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, “Towards a better understanding of context and context-awareness”, in *Handheld and Ubiquitous Computing: First International Symposium, HUC’99 Karlsruhe, Germany, September 27–29, 1999 Proceedings*, H.-W. Gellersen, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 304–307, ISBN: 978-3-540-48157-7. DOI: `10.1007/3-540-48157-5_29`. [Online]. Available: `https://doi.org/10.1007/3-540-48157-5_29`.

[6] S.-H. Park, Y.-J. Han, and T.-M. Chung, “Context-role based access control for context-aware application”, in *High Performance Computing and Communications*, M. Gerndt and D. Kranzlmüller, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 572–580, ISBN: 978-3-540-39372-6.

[7] R. Bhatti, E. Bertino, and A. Ghafoor, “A trust-based context-aware access control model for web-services”, in *Proceedings of the IEEE International Conference on Web Services*, ser. ICWS ’04, Washington, DC, USA: IEEE Computer Society, 2004, pp. 184–, ISBN: 0-7695-2167-3. DOI: `10.1109/ICWS.2004.15`. [Online]. Available: `https://doi.org/10.1109/ICWS.2004.15`.

[8] R. J. Hulsebosch, A. H. Salden, M. S. Bargh, P. W. G. Ebben, and J. Reitsma, “Context sensitive access control”, in *Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies*, ser. SACMAT ’05, Stockholm, Sweden: ACM, 2005, pp. 111–119, ISBN: 1-59593-045-0. DOI: `10.1145/1063979.1064000`. [Online]. Available: `http://doi.acm.org/10.1145/1063979.1064000`.

[9] B. M. Leiner, V. G. Cerf, D. D. Clark, R. E. Kahn, L. Kleinrock, D. C. Lynch, J. Postel, L. G. Roberts, and S. Wolff, “A brief history of the internet”, *SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 5, pp. 22–31, Oct. 2009, ISSN: 0146-4833. DOI: `10.1145/1629607.1629613`. [Online]. Available: `http://doi.acm.org/10.1145/1629607.1629613`.

[10] C.-L. Hsu, H.-P. Lu, and H.-H. Hsu, “Adoption of the mobile internet: An empirical study of multimedia message service (mms)”, *Omega*, vol. 35, no. 6, pp. 715–726, 2007, Special Issue on Telecommunications Applications, ISSN: 0305-0483. DOI: `https://doi.org/10.1016/j.omega.2006.03.005`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S0305048306000594`.

[11] G. Wu, S. Talwar, K. Johnsson, N. Himayat, and K. D. Johnson, “M2m: From mobile to embedded internet”, *IEEE Communications Magazine*, vol. 49, no. 4, pp. 36–43, Apr. 2011, ISSN: 0163-6804. DOI: `10.1109/MCOM.2011.5741144`.

[12] Cisco Systems, Inc., “Cisco annual internet report (2018–2023)”, Tech. Rep., 2020, Accessed on 20.9.2020. [Online]. Available: `https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html`.

[13] Gartner Inc., “Gartner identifies top 10 strategic iot technologies and trends”, Tech. Rep., Nov. 2018. [Online]. Available: `https://www.gartner.com/en/newsroom/press-releases/2018-11-07-gartner-identifies-top-10-strategic-iot-technologies-and-trends`.

[14] Intel Corporation, “Guide to internet of things”, Tech. Rep., 2020, Accessed on 20.9.2020. [Online]. Available: `https://www.intel.com/content/www/us/en/internet-of-things/infographics/guide-to-iot.html`.

[15] MarketsandMarkets Research Private Ltd., “Iot solutions and services market by component (platform, solution and services), service (consulting, and integration and deplpyment), vertical (smart manufacturing, smart energy and smart transportation), and region - global forecast to 2024”, Tech. Rep., 2020, Accessed on 25.10.2020. [Online]. Available: `https://www.marketsandmarkets.com/Market-Reports/iot-solutions-and-services-market-120466720.html`.

[16] N. Dragoni, S. Giallorenzo, A. L. Lafuente, M. Mazzara, F. Montesi, R. Mustafin, and L. Safina, “Microservices: Yesterday, today, and tomorrow”, in *Present and Ulterior Software Engineering*, M. Mazzara and B. Meyer, Eds. Cham: Springer International Publishing, 2017, pp. 195–216, ISBN: 978-3-319-67425-4. DOI: 10.1007/978-3-319-67425-4_12. [Online]. Available: https://doi.org/10.1007/978-3-319-67425-4_12.

[17] D. S. Markovic, D. Zivkovic, I. Branovic, R. Popovic, and D. Cvetkovic, “Smart power grid and cloud computing”, *Renewable and Sustainable Energy Reviews*, vol. 24, pp. 566–577, 2013, ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2013.03.068. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S136403211300227X.

[18] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, “Smart Grid Technologies: Communication Technologies and Standards”, *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, vol. 7, no. 4, 529–539, Nov. 2011, ISSN: 1551-3203. DOI: {10.1109/TII.2011.2166794}.

[19] X. Fang, S. Misra, G. Xue, and D. Yang, “Smart Grid - The New and Improved Power Grid: A Survey”, *IEEE COMMUNICATIONS SURVEYS AND TUTORIALS*, vol. 14, no. 4, 944–980, 2012. DOI: {10.1109/SURV.2011.101911.00087}.

[20] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, “A Survey on Smart Grid Potential Applications and Communication Requirements”, *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, vol. 9, no. 1, 28–42, Feb. 2013, ISSN: 1551-3203. DOI: {10.1109/TII.2012.2218253}.

[21] R. Morello, C. De Capua, G. Fulco, and S. C. Mukhopadhyay, “A smart power meter to monitor energy flow in smart grids: The role of advanced sensing and iot in the electric grid of the future”, *IEEE Sensors Journal*, vol. 17, no. 23, pp. 7828–7837, 2017.

[22] S. B. Baker, W. Xiang, and I. Atkinson, “Internet of things for smart healthcare: Technologies, challenges, and opportunities”, *IEEE Access*, vol. 5, pp. 26 521–26 544, 2017, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2017.2775180.

[23] A. Pantelopoulos and N. G. Bourbakis, “A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis”, *IEEE TRANSACTIONS ON SYSTEMS MAN AND CYBERNETICS PART C-APPLICATIONS AND REVIEWS*, vol. 40, no. 1, 1–12, Jan. 2010, ISSN: 1094-6977. DOI: {10.1109/TSMCC.2009.2032660}.

[24] Y.-L. Zheng, X.-R. Ding, C. C. Y. Poon, B. P. L. Lo, H. Zhang, X.-L. Zhou, G.-Z. Yang, N. Zhao, and Y.-T. Zhang, “Unobtrusive Sensing and Wearable Devices for Health Informatics”, *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, vol. 61, no. 5, SI, 1538–1554, May 2014, ISSN: 0018-9294. DOI: {10.1109/TBME.2014.2309951}.

[25] M. M. E. Mahmoud, J. J. P. C. Rodrigues, S. H. Ahmed, S. C. Shah, J. F. Al-Muhtadi, V. V. Korotaev, and V. H. C. De Albuquerque, "Enabling technologies on cloud of things for smart healthcare", *IEEE Access*, vol. 6, pp. 31 950–31 967, 2018.

[26] A. Solanas, C. Patsakis, M. Conti, *et al.*, "Smart health: A context-aware health paradigm within smart cities", *IEEE Communications Magazine*, vol. 52, no. 8, pp. 74–81, Aug. 2014, ISSN: 0163-6804. DOI: 10.1109/MCOM.2014.6871673.

[27] T. Nam and T. A. Pardo, "Conceptualizing smart city with dimensions of technology, people, and institutions", in *Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times*, ser. dg.o '11, College Park, Maryland, USA: Association for Computing Machinery, 2011, pp. 282–291, ISBN: 9781450307628. DOI: 10.1145/2037556.2037602. [Online]. Available: https://doi.org/10.1145/2037556.2037602.

[28] V. Albino, U. Berardi, and R. M. Dangelico, "Smart Cities: Definitions, Dimensions, Performance, and Initiatives", *JOURNAL OF URBAN TECHNOLOGY*, vol. 22, no. 1, 3–21, Jan. 2015, ISSN: 1063-0732. DOI: {10.1080/10630732.2014.942092}.

[29] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities", *IEEE INTERNET OF THINGS JOURNAL*, vol. 1, no. 1, 22–32, Feb. 2014, ISSN: 2327-4662. DOI: {10.1109/JIOT.2014.2306328}.

[30] M. Batty, K. W. Axhausen, F. Giannotti, A. Pozdnoukhov, A. Bazzani, M. Wachowicz, G. Ouzounis, and Y. Portugali, "Smart cities of the future", *EUROPEAN PHYSICAL JOURNAL-SPECIAL TOPICS*, vol. 214, no. 1, 481–518, Nov. 2012, ISSN: 1951-6355. DOI: {10.1140/epjst/e2012-01703-3}.

[31] C. Benevolo, R. P. Dameri, and B. D'Auria, "Smart mobility in smart city", in *Empowering Organizations*, T. Torre, A. M. Braccini, and R. Spinelli, Eds., Cham: Springer International Publishing, 2016, pp. 13–28, ISBN: 978-3-319-23784-8.

[32] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular Social Networks: Enabling Smart Mobility", *IEEE COMMUNICATIONS MAGAZINE*, vol. 55, no. 5, 49–55, May 2017, ISSN: 0163-6804. DOI: {10.1109/MCOM.2017.1600263}.

[33] A. Meijer, "Smart city governance: A local emergent perspective", in *Smarter as the New Urban Agenda: A Comprehensive View of the 21st Century City*, J. R. Gil-Garcia, T. A. Pardo, and T. Nam, Eds. Cham: Springer International Publishing, 2016, pp. 73–85, ISBN: 978-3-319-17620-8. DOI: 10.1007/978-3-319-17620-8_4. [Online]. Available: https://doi.org/10.1007/978-3-319-17620-8_4.

[34] G. V. Pereira, P. Parycek, E. Falco, and R. Kleinhans, "Smart governance in the context of smart cities: A literature review", *INFORMATION POLITY*, vol. 23, no. 2, 143–162, 2018, ISSN: 1570-1255. DOI: {10.3233/IP-170067}.

[35] A. Meijer and M. P. Rodriguez Bolivar, "Governing the smart city: a review of the literature on smart urban governance", *INTERNATIONAL REVIEW OF ADMINISTRATIVE SCIENCES*, vol. 82, no. 2, SI, 392–408, Jun. 2016, ISSN: 0020-8523. DOI: {10.1177/0020852314564308}.

[36] M. R. Alam, M. B. I. Reaz, and M. A. M. Ali, "A review of smart homes—past, present, and future", *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1190–1203, Nov. 2012, ISSN: 1558-2442. DOI: `10.1109/TSMCC.2012.2189204`.

[37] M. Chan, D. Esteve, C. Escriba, and E. Campo, "A review of smart homes - Present state and future challenges", *COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE*, vol. 91, no. 1, 55–81, Jun. 2008, ISSN: 0169-2607. DOI: `{10.1016/j.cmpb.2008.02.001}`.

[38] R. Sandhu, "Access control: The neglected frontier", in *Information Security and Privacy*, J. Pieprzyk and J. Seberry, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, pp. 219–227, ISBN: 978-3-540-49583-3.

[39] D. Ferraiolo, J. Cugini, and D. R. Kuhn, "Role-based access control (rbac): Features and motivations", in *Proceedings of 11th annual computer security application conference*, 1995, pp. 241–48.

[40] M. A. Al-Kahtani and R. Sandhu, "A model for attribute-based user-role assignment", in *18th Annual Computer Security Applications Conference, 2002. Proceedings.*, 2002, pp. 353–362. DOI: `10.1109/CSAC.2002.1176307`.

[41] M. Ge and S. L. Osborn, "A design for parameterized roles", in *Research Directions in Data and Applications Security XVIII*, C. Farkas and P. Samarati, Eds., Boston, MA: Springer US, 2004, pp. 251–264, ISBN: 978-1-4020-8128-6.

[42] J. Fischer, D. Marino, R. Majumdar, and T. Millstein, "Fine-grained access control with object-sensitive roles", in *ECOOP 2009 – Object-Oriented Programming*, S. Drossopoulou, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 173–194, ISBN: 978-3-642-03013-0.

[43] G. Sladic, B. Milosavljević, and Z. Konjovic, "Context-sensitive access control model for business processes", vol. 10, pp. 939–972, Jun. 2013.

[44] X. Jin, R. Krishnan, and R. Sandhu, "A unified attribute-based access control model covering dac, mac and rbac", in *Data and Applications Security and Privacy XXVI*, N. Cuppens-Boulahia, F. Cuppens, and J. Garcia-Alfaro, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 41–55, ISBN: 978-3-642-31540-4.

[45] K. Zeilenga, "Lightweight directory access protocol (ldap): Technical specification road map", RFC Editor, RFC 4510, Jun. 2006, `http://www.rfc-editor.org/rfc/rfc4510.txt`. [Online]. Available: `http://www.rfc-editor.org/rfc/rfc4510.txt`.

[46] N. Sakimura, J. Bradley, M. Jones, B. De Medeiros, and C. Mortimore, *Openid connect core 1.0 incorporating errata set 1*, 2014. [Online]. Available: `https://openid.net/connect/`.

[47] A. K. Dey, "Understanding and using context", *Personal Ubiquitous Comput.*, vol. 5, no. 1, pp. 4–7, Jan. 2001, ISSN: 1617-4909. DOI: `10.1007/s007790170019`. [Online]. Available: `http://dx.doi.org/10.1007/s007790170019`.

[48] G. Chen and D. Kotz, "A survey of context-aware mobile computing research", Hanover, NH, USA, Tech. Rep., 2000.

[49] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, “The anatomy of a context-aware application”, *Wirel. Netw.*, vol. 8, no. 2/3, pp. 187–197, Mar. 2002, ISSN: 1022-0038. DOI: 10.1023/A:1013767926256. [Online]. Available: http://dx.doi.org/10.1023/A:1013767926256.

[50] T. O’Reilly, “What is web 2.0? design patterns and business models for the next generation of software.”, 2005. [Online]. Available: http://oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html.

[51] N. Mallat, M. Rossi, V. K. Tuunainen, and A. Öörni, “The impact of use context on mobile services acceptance: The case of mobile ticketing”, *Information & Management*, vol. 46, no. 3, pp. 190–195, 2009, ISSN: 0378-7206. DOI: https://doi.org/10.1016/j.im.2008.11.008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378720609000202.

[52] T. Strang and C. Linnhoff-Popien, “A context modeling survey”, in *Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing, Nottingham/England*, 2004. [Online]. Available: http://pace.itee.uq.edu.au/cw2004/Paper15.pdf.

[53] A. W. ter Mors, C. Witteveen, J. Zutt, and F. A. Kuipers, “Context-aware route planning”, in *Multiagent System Technologies*, J. Dix and C. Witteveen, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 138–149, ISBN: 978-3-642-16178-0.

[54] M. Baldauf, S. Dustdar, and F. Rosenberg, “A survey on context-aware systems”, *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 2, no. 4, pp. 263–277, Jun. 2007, ISSN: 1743-8225. DOI: 10.1504/IJAHUC.2007.014070. [Online]. Available: http://dx.doi.org/10.1504/IJAHUC.2007.014070.

[55] L. Lin, T. Liu, S. Li, C. M. Sarathchandra Magurawalage, and S. Tu, “Priguarder: A privacy-aware access control approach based on attribute fuzzy grouping in cloud environments”, *IEEE Access*, vol. 6, pp. 1882–1893, 2018.

[56] M. J. Moyer and M. Abamad, “Generalized role-based access control”, in *Proceedings 21st International Conference on Distributed Computing Systems*, 2001, pp. 391–398.

[57] M. J. Covington, W. Long, S. Srinivasan, A. K. Dev, M. Ahamad, and G. D. Abowd, “Securing context-aware applications using environment roles”, in *Proceedings of the Sixth ACM Symposium on Access Control Models and Technologies*, ser. SACMAT ’01, Chantilly, Virginia, USA: ACM, 2001, pp. 10–20, ISBN: 1-58113-350-2. DOI: 10.1145/373256.373258. [Online]. Available: http://doi.acm.org/10.1145/373256.373258.

[58] D. Kulkarni and A. Tripathi, “Context-aware role-based access control in pervasive computing systems”, in *Proceedings of the 13th ACM Symposium on Access Control Models and Technologies*, ser. SACMAT ’08, Estes Park, CO, USA: Association for Computing Machinery, 2008, pp. 113–122, ISBN: 9781605581293. DOI: 10.1145/1377836.1377854. [Online]. Available: https://doi.org/10.1145/1377836.1377854.

[59] G. Neumann and M. Strembeck, “An approach to engineer and enforce context constraints in an rbac environment”, in *Proceedings of the Eighth ACM Symposium on Access Control Models and Technologies*, ser. SACMAT ’03, Como, Italy: Association for Computing Machinery, 2003, pp. 65–79, ISBN: 1581136811. DOI: 10.1145/775412.775421. [Online]. Available: https://doi.org/10.1145/775412.775421.

[60] G. K. Mostéfaoui and P. Brézillon, “A generic framework for context-based distributed authorizations”, in *Modeling and Using Context*, P. Blackburn, C. Ghidini, R. M. Turner, and F. Giunchiglia, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 204–217, ISBN: 978-3-540-44958-4.

[61] J. C. D. Lima, C. C. Rocha, I. Augustin, and M. A. R. Dantas, “A context-aware recommendation system to behavioral based authentication in mobile and pervasive environments”, in *2011 IFIP 9th International Conference on Embedded and Ubiquitous Computing*, 2011, pp. 312–319.

[62] A. Corrad, R. Montanari, and D. Tibaldi, “Context-based access control management in ubiquitous environments”, in *Third IEEE International Symposium on Network Computing and Applications, 2004. (NCA 2004). Proceedings.*, 2004, pp. 253–260.

[63] M. M. Molla and S. I. Ahamed, “A survey of middleware for sensor network and challenges”, in *2006 International Conference on Parallel Processing Workshops (ICPPW’06)*, 2006, 6 pp.–228.

[64] H. Chen, T. Finin, Anupam Joshi, L. Kagal, F. Perich, and Dipanjan Chakraborty, “Intelligent agents meet the semantic web in smart spaces”, *IEEE Internet Computing*, vol. 8, no. 6, pp. 69–79, 2004.

[65] M. Perttunen, J. Riekki, and O. Lassila, “Context representation and reasoning in pervasive computing: A review”, *International Journal of Multimedia and Ubiquitous Engineering*, pp. 1–28,

[66] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, and D. Riboni, “A survey of context modelling and reasoning techniques”, *Pervasive and Mobile Computing*, vol. 6, no. 2, pp. 161–180, 2010, Context Modelling, Reasoning and Management, ISSN: 1574-1192. DOI: https://doi.org/10.1016/j.pmcj.2009.06.002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1574119209000510.

[67] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, “Context aware computing for the internet of things: A survey”, *IEEE Communications Surveys Tutorials*, vol. 16, no. 1, pp. 414–454, First 2014, ISSN: 1553-877X. DOI: 10.1109/SURV.2013.042313.00197.

[68] K. Petersen, S. Vakkalanka, and L. Kuzniarz, “Guidelines for conducting systematic mapping studies in software engineering: An update”, *Information and Software Technology*, vol. 64, no. Supplement C, pp. 1–18, 2015, ISSN: 0950-5849. DOI: https://doi.org/10.1016/j.infsof.2015.03.007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950584915000646.

[69] M. binti Mohamad Noor and W. H. Hassan, "Current research on internet of things (iot) security: A survey", *Computer Networks*, vol. 148, pp. 283–294, 2019, ISSN: 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2018.11.025. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128618307035.

[70] P. M. Chanal and M. S. Kakkasageri, "Security and privacy in iot: A survey", *Wireless Personal Communications*, vol. 115, no. 2, pp. 1667–1693, Nov. 2020, ISSN: 1572-834X. DOI: 10.1007/s11277-020-07649-9. [Online]. Available: https://doi.org/10.1007/s11277-020-07649-9.

[71] N. Miloslavskaya and A. Tolstoy, "Internet of things: Information security challenges and solutions", *Cluster Computing*, vol. 22, no. 1, pp. 103–119, Mar. 2019, ISSN: 1573-7543. DOI: 10.1007/s10586-018-2823-6. [Online]. Available: https://doi.org/10.1007/s10586-018-2823-6.

[72] F. H. Al-Naji and R. Zagrouba, "A survey on continuous authentication methods in internet of things environment", *Computer Communications*, vol. 163, pp. 109–133, 2020, ISSN: 0140-3664. DOI: https://doi.org/10.1016/j.comcom.2020.09.006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366420319204.

[73] K. Tange, M. De Donno, X. Fafoutis, and N. Dragoni, "A systematic survey of industrial internet of things security: Requirements and fog computing opportunities", *IEEE Communications Surveys Tutorials*, vol. 22, no. 4, pp. 2489–2520, 2020. DOI: 10.1109/COMST.2020.3011208.

[74] S. Rose, D. Engel, N. Cramer, and W. Cowley, "Automatic keyword extraction from individual documents", *Text Mining: Applications and Theory*, pp. 1–20, 2010. DOI: 10.1002/9780470689646.ch1.

[75] *Pdftotext*. [Online]. Available: http://www.xpdfreader.com.

[76] I. Agadakos, P. Hallgren, D. Damopoulos, A. Sabelfeld, and G. Portokalidis, "Location-enhanced authentication using the iot: Because you cannot be in two places at once", in *Proceedings of the 32Nd Annual Conference on Computer Security Applications*, ser. ACSAC '16, Los Angeles, California, USA: ACM, 2016, pp. 251–264, ISBN: 978-1-4503-4771-6. DOI: 10.1145/2991079.2991090. [Online]. Available: http://doi.acm.org/10.1145/2991079.2991090.

[77] G. Alpár, L. Batina, L. Batten, V. Moonsamy, A. Krasnova, A. Guellier, and I. Natgunanathan, "New directions in iot privacy using attribute-based authentication", in *Proceedings of the ACM International Conference on Computing Frontiers*, ser. CF '16, Como, Italy: ACM, 2016, pp. 461–466, ISBN: 978-1-4503-4128-8. DOI: 10.1145/2903150.2911710. [Online]. Available: http://doi.acm.org/10.1145/2903150.2911710.

[78] L. Barreto, A. Celesti, M. Villari, M. Fazio, and A. Puliafito, "An authentication model for iot clouds", in *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, Aug. 2015, pp. 1032–1035. DOI: 10.1145/2808797.2809361.

[79] M. Cagnazzo, M. Hertlein, and N. Pohlmann, "An usable application for authentication, communication and access management in the internet of things", in *Information and Software Technologies: 22nd International Conference, ICIST 2016, Druskininkai, Lithuania, October 13-15, 2016, Proceedings*. Cham: Springer International Publishing, 2016, pp. 722–731, ISBN: 978-3-319-46254-7. DOI: 10.1007/978-3-319-46254-7_58. [Online]. Available: https://doi.org/10.1007/978-3-319-46254-7_58.

[80] F. Chen, Y. Luo, J. Zhang, J. Zhu, Z. Zhang, C. Zhao, and T. Wang, "An infrastructure framework for privacy protection of community medical internet of things", *World Wide Web*, Apr. 2017, ISSN: 1573-1413. DOI: 10.1007/s11280-017-0455-z. [Online]. Available: https://doi.org/10.1007/s11280-017-0455-z.

[81] P. Fremantle, J. Kopecký, and B. Aziz, "Web api management meets the internet of things", in *The Semantic Web: ESWC 2015 Satellite Events: ESWC 2015 Satellite Events, Portorož, Slovenia, May 31 – June 4, 2015, Revised Selected Papers*. Cham: Springer International Publishing, 2015, pp. 367–375, ISBN: 978-3-319-25639-9. DOI: 10.1007/978-3-319-25639-9_49. [Online]. Available: https://doi.org/10.1007/978-3-319-25639-9_49.

[82] S. Gerdes, C. Bormann, and O. Bergmann, "Chapter 11 - keeping users empowered in a cloudy internet of things", in *The Cloud Security Ecosystem*, R. Ko and K.-K. R. Choo, Eds., Boston: Syngress, 2015, pp. 231–247, ISBN: 978-0-12-801595-7. DOI: https://doi.org/10.1016/B978-0-12-801595-7.00011-2. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128015957000112.

[83] J. L. Hernández-Ramos, M. P. Pawlowski, A. J. Jara, A. F. Skarmeta, and L. Ladid, "Toward a lightweight authentication and authorization framework for smart objects", *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 4, pp. 690–702, Apr. 2015, ISSN: 0733-8716. DOI: 10.1109/JSAC.2015.2393436.

[84] T. Kumar, A. Braeken, M. Liyanage, and M. Ylianttila, "Identity privacy preserving biometric based authentication scheme for naked healthcare environment", in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–7. DOI: 10.1109/ICC.2017.7996966.

[85] S.-H. Lee, K.-W. Huang, and C.-S. Yang, "Tbas: Token-based authorization service architecture in internet of things scenarios", *International Journal of Distributed Sensor Networks*, vol. 13, no. 7, p. 1 550 147 717 718 496, 2017. DOI: 10.1177/1550147717718496. eprint: https://doi.org/10.1177/1550147717718496. [Online]. Available: https://doi.org/10.1177/1550147717718496.

[86] L. Liu, B. Fang, and B. Yi, "A general framework of nonleakage-based authentication using csp for the internet of things", in *Web Technologies and Applications: APWeb 2014 Workshops, SNA, NIS, and IoTS, Changsha, China, September 5, 2014. Proceedings*. Cham: Springer International Publishing, 2014, pp. 312–324, ISBN: 978-3-319-11119-3. DOI: 10.1007/978-3-319-11119-3_29. [Online]. Available: https://doi.org/10.1007/978-3-319-11119-3_29.

[87] A. Pinto and R. Costa, "Hash-chain based authentication for iot devices and rest web-services", in *Ambient Intelligence- Software and Applications – 7th International Symposium on Ambient Intelligence (ISAmI 2016)*, H. Lindgren, J. F. De Paz, P. Novais, A. Fernández-Caballero, H. Yoe, A. Jiménez Ramírez, and G. Villarrubia, Eds. Cham: Springer International Publishing, 2016, pp. 189–196, ISBN: 978-3-319-40114-0. DOI: 10.1007/978-3-319-40114-0_21. [Online]. Available: https://doi.org/10.1007/978-3-319-40114-0_21.

[88] M. Shahzad and M. P. Singh, "Continuous authentication and authorization for the internet of things", *IEEE Internet Computing*, vol. 21, no. 2, pp. 86–90, Mar. 2017, ISSN: 1089-7801. DOI: 10.1109/MIC.2017.33.

[89] N. Shone, C. Dobbins, W. Hurst, and Q. Shi, "Digital memories based mobile user authentication for iot", in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, Oct. 2015, pp. 1796–1802. DOI: 10.1109/CIT/IUCC/DASC/PICOM.2015.270.

[90] Q. Tasali, C. Chowdhury, and E. Y. Vasserman, "A flexible authorization architecture for systems of interoperable medical devices", in *Proceedings of the 22Nd ACM on Symposium on Access Control Models and Technologies*, ser. SACMAT '17 Abstracts, Indianapolis, Indiana, USA: ACM, 2017, pp. 9–20, ISBN: 978-1-4503-4702-0. DOI: 10.1145/3078861.3078862. [Online]. Available: http://doi.acm.org/10.1145/3078861.3078862.

[91] S. Unger and D. Timmermann, "Dpwsec: Devices profile for web services security", in *2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, Apr. 2015, pp. 1–6. DOI: 10.1109/ISSNIP.2015.7106961.

[92] S. Wiseman, G. Soto Mino, A. L. Cox, S. J. Gould, J. Moore, and C. Needham, "Use your words: Designing one-time pairing codes to improve user experience", in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ser. CHI '16, San Jose, California, USA: ACM, 2016, pp. 1385–1389, ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858377. [Online]. Available: http://doi.acm.org/10.1145/2858036.2858377.

[93] S. Sicari, A. Rizzardi, L. Grieco, G. Piro, and A. Coen-Porisini, "A policy enforcement framework for internet of things applications in the smart health", *Smart Health*, vol. 3-4, pp. 39–74, 2017, ISSN: 2352-6483. DOI: https://doi.org/10.1016/j.smhl.2017.06.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2352648316300435.

[94] A. Outchakoucht, H. ES-SAMAALI, and J. Philippe, "Dynamic access control policy based on blockchain and machine learning for the internet of things", *International Journal of Advanced Computer Science and Applications*, vol. 8, Jan. 2017. DOI: 10.14569/IJACSA.2017.080757. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2017.080757.

[95] N. YE, Y. Zhu, R.-c. WANG, R. Malekian, and L. Qiao-min, "An efficient authentication and access control scheme for perception layer of internet of things", *Applied Mathematics & Information Sciences*, vol. 8, Jul. 2014.

[96] J. Bernal Bernabe, J. L. Hernandez-Ramos, and A. F. Skarmeta Gomez, "Holistic privacy-preserving identity management system for the internet of things", *Mobile Information Systems*, vol. 2017, p. 20, 2017. DOI: 10.1155/2017/6384186.

[97] B.-C. Chifor, I. Bica, V.-V. Patriciu, and F. Pop, "A security authorization scheme for smart home internet of things devices", *Future Generation Computer Systems*, vol. 86, pp. 740–749, 2018, ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2017.05.048. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17311020.

[98] W. Chiu, C. Su, C.-Y. Fan, C.-M. Chen, and K.-H. Yeh, "Authentication with what you see and remember in the internet of things", *Symmetry*, vol. 10, no. 11, p. 537, Oct. 2018, ISSN: 2073-8994. DOI: 10.3390/sym10110537. [Online]. Available: http://dx.doi.org/10.3390/sym10110537.

[99] F. Sun, C. Mao, X. Fan, and Y. Li, "Accelerometer-based speed-adaptive gait authentication method for wearable iot devices", *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 820–830, 2019. DOI: 10.1109/JIOT.2018.2860592.

[100] S.-R. Oh, Y.-G. Kim, and S. Cho, "An interoperable access control framework for diverse iot platforms based on oauth and role", *Sensors*, vol. 19, no. 8, p. 1884, Apr. 2019, ISSN: 1424-8220. DOI: 10.3390/s19081884. [Online]. Available: http://dx.doi.org/10.3390/s19081884.

[101] H. Yan, Y. Wang, C. Jia, J. Li, Y. Xiang, and W. Pedrycz, "Iot-fbac: Function-based access control scheme using identity-based encryption in iot", *Future Generation Computer Systems*, vol. 95, pp. 344–353, 2019, ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2018.12.061. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X1830997X.

[102] P. Nespoli, M. Zago, A. Huertas Celdrán, M. Gil Pérez, F. Gómez Mármol, and F. J. García Clemente, "Palot: Profiling and authenticating users leveraging internet of things", *Sensors*, vol. 19, no. 12, p. 2832, Jun. 2019, ISSN: 1424-8220. DOI: 10.3390/s19122832. [Online]. Available: http://dx.doi.org/10.3390/s19122832.

[103] N. Ghosh, S. Chandra, V. Sachidananda, and Y. Elovici, "Softauthz: A context-aware, behavior-based authorization framework for home iot", *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 773–10 785, 2019. DOI: 10.1109/JIOT.2019.2941767.

[104] S. Batool, A. Hassan, N. A. Saqib, and M. A. K. Khattak, "Authentication of remote iot users based on deeper gait analysis of sensor data", *IEEE Access*, vol. 8, pp. 101 784–101 796, 2020. DOI: 10.1109/ACCESS.2020.2998412.

[105] G. Ali, N. Ahmad, Y. Cao, S. Khan, H. Cruickshank, E. A. Qazi, and A. Ali, "Xdbauth: Blockchain based cross domain authentication and authorization framework for internet of things", *IEEE Access*, vol. 8, pp. 58 800–58 816, 2020. DOI: 10.1109/ACCESS.2020.2982542.

[106] S.-R. Oh and Y.-G. Kim, "Afaas: Authorization framework as a service for internet of things based on interoperable oauth", *International Journal of Distributed Sensor Networks*, vol. 16, no. 2, p. 1 550 147 720 906 388, 2020. DOI: 10.1177/1550147720906388. eprint: https://doi.org/10.1177/1550147720906388. [Online]. Available: https://doi.org/10.1177/1550147720906388.

[107] U. Khalid, M. Asim, T. Baker, P. C. K. Hung, M. A. Tariq, and L. Rafferty, "A decentralized lightweight blockchain-based authentication mechanism for iot systems", *Cluster Computing*, vol. 23, no. 3, pp. 2067–2087, Sep. 2020, ISSN: 1573-7543. DOI: 10.1007/s10586-020-03058-6.

[108] S. Zhang, Y. Cao, Z. Ning, F. Xue, D. Cao, and Y. Yang, "A Heterogeneous IoT Node Authentication Scheme Based on Hybrid Blockchain and Trust Value", *KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS*, vol. 14, no. 9, 3615–3638, Sep. 2020, ISSN: 1976-7277. DOI: 10.3837/tiis.2020.09.003.

[109] K. N. Pallavi and V. Ravi Kumar, "Authentication-based access control and data exchanging mechanism of iot devices in fog computing environment", *Wireless Personal Communications*, Oct. 2020, ISSN: 1572-834X. DOI: 10.1007/s11277-020-07834-w. [Online]. Available: https://doi.org/10.1007/s11277-020-07834-w.

[110] A. Alkhresheh, K. Elgazzar, and H. S. Hassanein, "Daciot: Dynamic access control framework for iot deployments", *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11 401–11 419, 2020. DOI: 10.1109/JIOT.2020.3002709.

[111] J. Bernal Bernabe, J. L. Hernandez Ramos, and A. F. Skarmeta Gomez, "Taciot: Multidimensional trust-aware access control system for the internet of things", *Soft Computing*, vol. 20, no. 5, pp. 1763–1779, May 2016, ISSN: 1433-7479. DOI: 10.1007/s00500-015-1705-6. [Online]. Available: https://doi.org/10.1007/s00500-015-1705-6.

[112] S. Cirani and M. Picone, "Effective authorization for the web of things", in *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, Dec. 2015, pp. 316–320. DOI: 10.1109/WF-IoT.2015.7389073.

[113] W. Han, Y. Zhang, Z. Guo, and E. Bertino, "Fine-grained business data confidentiality control in cross-organizational tracking", in *Proceedings of the 20th ACM Symposium on Access Control Models and Technologies*, ser. SACMAT '15, Vienna, Austria: ACM, 2015, pp. 135–145, ISBN: 978-1-4503-3556-0. DOI: 10.1145/2752952.2752973. [Online]. Available: http://doi.acm.org/10.1145/2752952.2752973.

[114] A. Kurniawan and M. Kyas, "A trust model-based bayesian decision theory in large scale internet of things", in *2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, Apr. 2015, pp. 1–5. DOI: 10.1109/ISSNIP.2015.7106964.

[115] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman, "Towards a novel privacy-preserving access control model based on blockchain technology in iot", in *Europe and MENA Cooperation Advances in Information and Communication Technologies*, Á. Rocha, M. Serrhini, and C. Felgueiras, Eds. Cham: Springer International

Publishing, 2017, pp. 523–533, ISBN: 978-3-319-46568-5. DOI: 10.1007/978-3-319-46568-5_53. [Online]. Available: https://doi.org/10.1007/978-3-319-46568-5_53.

[116] P. Solapurkar, "Building secure healthcare services using oauth 2.0 and json web token in iot cloud scenario", in *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, Dec. 2016, pp. 99–104. DOI: 10.1109/IC3I.2016.7917942.

[117] S. Lee, J. Choi, J. Kim, B. Cho, S. Lee, H. Kim, and J. Kim, "Fact: Functionality-centric access control system for iot programming frameworks", in *Proceedings of the 22Nd ACM on Symposium on Access Control Models and Technologies*, ser. SACMAT '17 Abstracts, Indianapolis, Indiana, USA: ACM, 2017, pp. 43–54, ISBN: 978-1-4503-4702-0. DOI: 10.1145/3078861.3078864. [Online]. Available: http://doi.acm.org/10.1145/3078861.3078864.

[118] S. Bandara, T. Yashiro, N. Koshizuka, and K. Sakamura, "Access control framework for api-enabled devices in smart buildings", in *2016 22nd Asia-Pacific Conference on Communications (APCC)*, Aug. 2016, pp. 210–217. DOI: 10.1109/APCC.2016.7581479.

[119] A. Biason, C. Pielli, A. Zanella, and M. Zorzi, "Access control for iot nodes with energy and fidelity constraints", *IEEE Transactions on Wireless Communications*, pp. 1–1, 2018, ISSN: 1536-1276. DOI: 10.1109/TWC.2018.2808520.

[120] Q. Huang, L. Wang, and Y. Yang, "Decent: Secure and fine-grained data access control with policy updating for constrained iot devices", *World Wide Web*, vol. 21, no. 1, pp. 151–167, Jan. 2018, ISSN: 1573-1413. DOI: 10.1007/s11280-017-0462-0. [Online]. Available: https://doi.org/10.1007/s11280-017-0462-0.

[121] J. A. Martínez, J. L. Hernández-Ramos, V. Beltrán, A. Skarmeta, and P. M. Ruiz, "A user-centric internet of things platform to empower users for managing security and privacy concerns in the internet of energy", *International Journal of Distributed Sensor Networks*, vol. 13, no. 8, p. 1 550 147 717 727 974, 2017. DOI: 10.1177/1550147717727974. [Online]. Available: https://doi.org/10.1177/1550147717727974.

[122] S. Pal, T. Rabehaja, M. Hitchens, V. Varadharajan, and A. Hill, "On the design of a flexible delegation model for the internet of things using blockchain", *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3521–3530, 2020. DOI: 10.1109/TII.2019.2925898.

[123] C. Lupascu, A. Lupascu, and I. Bica, "Dlt based authentication framework for industrial iot devices", *Sensors*, vol. 20, no. 9, p. 2621, May 2020, ISSN: 1424-8220. DOI: 10.3390/s20092621. [Online]. Available: http://dx.doi.org/10.3390/s20092621.

[124] H. B. Djilali, D. Tandjaoui, and H. Khemissa, "Enhanced dynamic team access control for collaborative internet of things using context", *Transactions on Emerging Telecommunications Technologies*, vol. n/a, no. n/a, e4083, DOI: https://doi.org/10.1002/ett.4083. eprint: https://onlinelibrary.wiley.

com/doi/pdf/10.1002/ett.4083. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.4083.

[125] S. Gusmeroli, S. Piccione, and D. Rotondi, "A capability-based security approach to manage access control in the internet of things", *Mathematical and Computer Modelling*, vol. 58, no. 5, pp. 1189–1205, 2013, The Measurement of Undesirable Outputs: Models Development and Empirical Analyses and Advances in mobile, ubiquitous and cognitive computing, ISSN: 0895-7177. DOI: https://doi.org/10.1016/j.mcm.2013.02.006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S089571771300054X.

[126] A. Majeed and A. Al-Yasiri, "Formulating a global identifier based on actor relationship for the internet of things", in *Interoperability, Safety and Security in IoT: Second International Conference, InterIoT 2016 and Third International Conference, SaSeIoT 2016, Paris, France, October 26-27, 2016, Revised Selected Papers*. Cham: Springer International Publishing, 2017, pp. 79–91, ISBN: 978-3-319-52727-7. DOI: 10.1007/978-3-319-52727-7_10. [Online]. Available: https://doi.org/10.1007/978-3-319-52727-7_10.

[127] D. Schreckling, J. D. Parra, C. Doukas, and J. Posegga, "Data-centric security for the iot", in *Internet of Things. IoT Infrastructures: Second International Summit, IoT 360 2015, Rome, Italy, October 27-29, 2015, Revised Selected Papers, Part II*. Cham: Springer International Publishing, 2016, pp. 77–86, ISBN: 978-3-319-47075-7. DOI: 10.1007/978-3-319-47075-7_10. [Online]. Available: https://doi.org/10.1007/978-3-319-47075-7_10.

[128] K. Fysarakis, I. Papaefstathiou, C. Manifavas, K. Rantos, and O. Sultatos, "Policy-based access control for dpws-enabled ubiquitous devices", in *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, Sep. 2014, pp. 1–8. DOI: 10.1109/ETFA.2014.7005233.

[129] P. Fremantle and B. Aziz, "Cloud-based federated identity for the internet of things", *Annals of Telecommunications*, vol. 73, no. 7, pp. 415–427, Aug. 2018, ISSN: 1958-9395. DOI: 10.1007/s12243-018-0641-8. [Online]. Available: https://doi.org/10.1007/s12243-018-0641-8.

[130] D. Hussein, E. Bertin, and V. Frey, "A community-driven access control approach in distributed iot environments", *IEEE Communications Magazine*, vol. 55, no. 3, pp. 146–153, Mar. 2017, ISSN: 0163-6804. DOI: 10.1109/MCOM.2017.1600611CM.

[131] V. Sivaraman, H. H. Gharakheili, A. Vishwanath, R. Boreli, and O. Mehani, "Network-level security and privacy control for smart-home iot devices", in *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct. 2015, pp. 163–167. DOI: 10.1109/WiMOB.2015.7347956.

[132] M. Poulymenopoulou, F. Malamateniou, and G. Vassilacopoulos, "A virtual phr authorization system", in *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, Jun. 2014, pp. 73–76. DOI: 10.1109/BHI.2014.6864307.

[133] J. Wilson, R. S. Wahby, H. Corrigan-Gibbs, D. Boneh, P. Levis, and K. Winstein, "Trust but verify: Auditing the secure internet of things", in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '17, Niagara Falls, New York, USA: ACM, 2017, pp. 464–474, ISBN: 978-1-4503-4928-4. DOI: 10.1145/3081333.3081342. [Online]. Available: http://doi.acm.org/10.1145/3081333.3081342.

[134] I. B. .-.-. Pasquier, A. A. Ouahman, A. A. E. Kalam, and M. O. de Montfort, "Smartorbac security and privacy in the internet of things", in *2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA)*, Nov. 2015, pp. 1–8. DOI: 10.1109/AICCSA.2015.7507098.

[135] B. Gong, Y. Wang, X. Liu, F. Qi, and Z. Sun, "A trusted attestation mechanism for the sensing nodes of internet of things based on dynamic trusted measurement", *China Communications*, vol. 15, no. 2, pp. 100–121, 2018. DOI: 10.1109/CC.2018.8300276.

[136] H.-C. Chen, "Collaboration iot-based rbac with trust evaluation algorithm model for massive iot integrated application", *Mobile Networks and Applications*, vol. 24, no. 3, pp. 839–852, Jun. 2019, ISSN: 1572-8153. DOI: 10.1007/s11036-018-1085-0. [Online]. Available: https://doi.org/10.1007/s11036-018-1085-0.

[137] C.-Y. Chen, "Efficient authentication for tiered internet of things networks", in *Quality, Reliability, Security and Robustness in Heterogeneous Networks: 12th International Conference, QShine 2016, Seoul, Korea, July 7–8, 2016, Proceedings*. Cham: Springer International Publishing, 2017, pp. 469–472, ISBN: 978-3-319-60717-7. DOI: 10.1007/978-3-319-60717-7_46. [Online]. Available: https://doi.org/10.1007/978-3-319-60717-7_46.

[138] H. Ren, Y. Song, S. Yang, and F. Situ, "Secure smart home: A voiceprint and internet based authentication system for remote accessing", in *2016 11th International Conference on Computer Science Education (ICCSE)*, Aug. 2016, pp. 247–251. DOI: 10.1109/ICCSE.2016.7581588.

[139] N. Pohlmann, M. Hertlein, and P. Manaras, "Bring your own device for authentication (byod4a) – the xign–system", in *ISSE 2015: Highlights of the Information Security Solutions Europe 2015 Conference*, H. Reimer, N. Pohlmann, and W. Schneider, Eds. Wiesbaden: Springer Fachmedien Wiesbaden, 2015, pp. 240–250, ISBN: 978-3-658-10934-9. DOI: 10.1007/978-3-658-10934-9_20. [Online]. Available: https://doi.org/10.1007/978-3-658-10934-9_20.

[140] A. Bassi, M. Bauer, M. Fiedler, T. Kramp, R. van Kranenburg, S. Lange, and S. Meissner, *Enabling Things to Talk: Designing IoT Solutions with the IoT Architectural Reference Model*, 1st. Springer Publishing Company, Incorporated, 2016, ISBN: 3662524945, 9783662524947.

[141] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz, "Extensible authentication protocol (eap)", RFC Editor, RFC 3748, Jun. 2004, http://www.rfc-editor.org/rfc/rfc3748.txt. [Online]. Available: http://www.rfc-editor.org/rfc/rfc3748.txt.

[142] R. T. Fielding, "Architectural styles and the design of network-based software architectures", in. University of California, 2000, ch. Representational State Transfer (REST).

[143] E. Rescorla, "Http over tls", RFC Editor, RFC 2818, May 2000, `http://www.rfc-editor.org/rfc/rfc2818.txt`. [Online]. Available: `http://www.rfc-editor.org/rfc/rfc2818.txt`.

[144] A. W. Roscoe, *The Theory and Practice of Concurrency*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1997, ISBN: 0136744095.

[145] J. Camenisch and E. Van Herreweghen, "Design and implementation of the idemix anonymous credential system", in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, ser. CCS '02, Washington, DC, USA: ACM, 2002, pp. 21–30, ISBN: 1-58113-612-9. DOI: `10.1145/586110.586114`. [Online]. Available: `http://doi.acm.org/10.1145/586110.586114`.

[146] S. Cirani, M. Picone, P. Gonizzi, L. Veltri, and G. Ferrari, "Iot-oas: An oauth-based authorization service architecture for secure services in iot scenarios", *IEEE Sensors Journal*, vol. 15, no. 2, pp. 1224–1234, Feb. 2015, ISSN: 1530-437X.

[147] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption", in *2007 IEEE Symposium on Security and Privacy (SP '07)*, 2007, pp. 321–334. DOI: `10.1109/SP.2007.11`.

[148] D. Boneh and X. Boyen, "Efficient selective-id secure identity-based encryption without random oracles", in *Advances in Cryptology - EUROCRYPT 2004*, C. Cachin and J. L. Camenisch, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 223–238, ISBN: 978-3-540-24676-3.

[149] *The extensible access control markup language (xacml) version 3.0*, OASIS Standard, 2017. [Online]. Available: `http://openid.net/developers/specs/`.

[150] D. Hardt, *The OAuth 2.0 Authorization Framework*, RFC 6749 (Proposed Standard), Internet Engineering Task Force, Oct. 2012. [Online]. Available: `http://www.ietf.org/rfc/rfc6749.txt`.

[151] M. Jones, J. Bradley, and N. Sakimura, *JSON Web Token (JWT)*, RFC 7519 (Proposed Standard), Updated by RFC 7797, Internet Engineering Task Force, May 2015. [Online]. Available: `http://www.ietf.org/rfc/rfc7519.txt`.

[152] A. A. E. Kalam, R. E. Baida, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarte, A. Miege, C. Saurel, and G. Trouessin, "Organization based access control", in *Proceedings POLICY 2003. IEEE 4th International Workshop on Policies for Distributed Systems and Networks*, Jun. 2003, pp. 120–131. DOI: `10.1109/POLICY.2003.1206966`.

[153] *Astm f2761-09(2013), medical devices and medical systems - essential safety requirements for equipment comprising the patient-centric integrated clinical environment (ice) - part 1: General requirements and conceptual model*, ASTM International, West Conshohocken, PA, 2009. [Online]. Available: `www.astm.org`.

[154] J. Hatcliff, A. King, I. Lee, A. Macdonald, A. Fernando, M. Robkin, E. Vasserman, S. Weininger, and J. M. Goldman, "Rationale and architecture principles for medical application platforms", in *2012 IEEE/ACM Third International Conference on Cyber-Physical Systems*, Apr. 2012, pp. 3–12. DOI: `10.1109/ICCPS.2012.9`.

[155] R. Lindemann, D. Baghdasaryan, and E. Tiffany, *Fido universal authentication framework protocol, version v1. 0-rd-20140209*, FIDO Alliance, Feb. 2014.

[156] *Web services security: Soap message security 1.1*, OASIS Standard, 2006. [Online]. Available: `https://www.oasis-open.org/committees/wss/`.

[157] B. C. Neuman, "Proxy-based authorization and accounting for distributed systems", in *[1993] Proceedings. The 13th International Conference on Distributed Computing Systems*, May 1993, pp. 283–291. DOI: `10.1109/ICDCS.1993.287698`.

[158] *Google scholar*, `https://scholar.google.com/`.

[159] G. E, "The history and meaning of the journal impact factor", *JAMA*, vol. 295, no. 1, pp. 90–93, 2006. DOI: `10.1001/jama.295.1.90`. eprint: `/data/journals/jama/5006/jco50055.pdf`. [Online]. Available: `+%20http://dx.doi.org/10.1001/jama.295.1.90`.

[160] *Core conference ranking*, `http://portal.core.edu.au/conf-ranks/`.

[161] R. L. Lawrence and A. Wright, "Rule-based classification systems using classification and regression tree (cart) analysis", *Photogrammetric engineering and remote sensing*, vol. 67, no. 10, pp. 1137–1142, 2001.

[162] K. Nozaki, H. Ishibuchi, and H. Tanaka, "Adaptive fuzzy rule-based classification systems", *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 3, pp. 238–250, Aug. 1996, ISSN: 1063-6706. DOI: `10.1109/91.531768`.

[163] "Information technology – open systems interconnection – basic reference model: Naming and addressing", International Organization for Standardization and the International Electrotechnical Commission, Geneva, Switzerland, ISO/EIC 7498-1:1997, 1997.

[164] L. Oliveira, J. Rodrigues, S. Kozlov, R. Rabêlo, and V. Albuquerque, "Mac layer protocols for internet of things: A survey", *Future Internet*, vol. 11, no. 1, p. 16, Jan. 2019, ISSN: 1999-5903. DOI: `10.3390/fi11010016`. [Online]. Available: `http://dx.doi.org/10.3390/fi11010016`.

[165] *Picketlink*. [Online]. Available: `http://picketlink.org/`.

[166] *Keycloak*. [Online]. Available: `https://www.keycloak.org/`.

[167] *Java platform, enterprise edition (java ee) 7*. [Online]. Available: `https://docs.oracle.com/javaee/7/index.html`.

[168] P. Tarr, H. Ossher, W. Harrison, and S. M. Sutton, "<i>n</i> degrees of separation: Multi-dimensional separation of concerns", in *Proceedings of the 21st International Conference on Software Engineering*, ser. ICSE '99, Los Angeles, California, USA: Association for Computing Machinery, 1999, pp. 107–119, ISBN: 1581130740. DOI: `10.1145/302405.302457`. [Online]. Available: `https://doi.org/10.1145/302405.302457`.

[169] W. P. Stevens, G. J. Myers, and L. L. Constantine, "Structured design", *IBM Systems Journal*, vol. 13, no. 2, pp. 115–139, 1974. DOI: 10.1147/sj.132.0115.

[170] R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed internet of things", *Computer Networks*, vol. 57, no. 10, pp. 2266–2279, 2013, Towards a Science of Cyber SecuritySecurity and Identity Architecture for the Future Internet, ISSN: 1389-1286. DOI: http://dx.doi.org/10.1016/j.comnet.2012.12.018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128613000054.

[171] Q. Jing, A. V. Vasilakos, J. Wan, J. Lu, and D. Qiu, "Security of the internet of things: Perspectives and challenges", *Wireless Networks*, vol. 20, no. 8, pp. 2481–2501, 2014, ISSN: 1572-8196. DOI: 10.1007/s11276-014-0761-7. [Online]. Available: http://dx.doi.org/10.1007/s11276-014-0761-7.

[172] R. Fielding and J. Reschke, *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content*, RFC 7231 (Proposed Standard), Internet Engineering Task Force, Jun. 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7231.txt.

[173] A. Banks, E. Briggs, K. Borgendale, and R. Gupta, "MQTT Version 5.0", OASIS, Standard, Mar. 2019. [Online]. Available: https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html.

[174] A. Freier, P. Karlton, and P. Kocher, *The Secure Sockets Layer (SSL) Protocol Version 3.0*, RFC 6101 (Historic), Internet Engineering Task Force, Aug. 2011. [Online]. Available: http://www.ietf.org/rfc/rfc6101.txt.

[175] A. Finkelsteiin and J. Kramer, "Software engineering: A roadmap", in *Proceedings of the Conference on The Future of Software Engineering*, ser. ICSE '00, Limerick, Ireland: ACM, 2000, pp. 3–22, ISBN: 1-58113-253-0. DOI: 10.1145/336512.336519. [Online]. Available: http://doi.acm.org/10.1145/336512.336519.

[176] *Shibboleth.* [Online]. Available: https://www.shibboleth.net/.

[177] M. Jones and D. Hardt, "The oauth 2.0 authorization framework: Bearer token usage", RFC Editor, RFC 6750, Oct. 2012, http://www.rfc-editor.org/rfc/rfc6750.txt. [Online]. Available: http://www.rfc-editor.org/rfc/rfc6750.txt.

Scientific results of author

This section shows publications and a list of selected citations of the author. Where applicable, I list Impact Factor (2019) or CORE2020 ranking and citation count. All the citations are obtained from the Google Scholar database on March 3rd, 2021, and exclude auto citations.

Awards

During my studies, I have received a Fulbright scholarship, which allowed me to spend a fruitful year full of exciting research on Baylor University in Waco, Texas, USA.

Related Publications

Journals with Impact Factor

[A.1] M. Trnka, J. Svacina, T. Cerny, E. Song, J. Hong, and M. Bures, "Securing internet of things devices using the network context", *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4017–4027, 2020, ISSN: 1551-3203. DOI: `10.1109/TII.2019.2954100` (IF: 9.11, Citations: 1, Contribution: 50%)
Selected citation: [B.5]

[A.2] M. Trnka, T. Cerny, and N. Stickney, "Survey of authentication and authorization for the internet of things", *Security and Communication Networks*, vol. 2018, pp. 1–17, 2018, ISSN: 1939-0114. DOI: `10.1155/2018/4351603` (IF: 1.29, Citations: 29, Contribution: 90%)
Selected citations: [B.1], [B.3], [B.4], [B.6], [B.8], [B.11], [B.16], [B.17],[B.21]

[A.3] (Under review) M. Bures, K. Frajtak, V. Rechtberger, M. Klima, M. Trnka, X. Bellekens, T. Cerny, and B. S. Ahmed, "Code-quality Metrics for IoT Systems", *Computer Science and Information Systems*, 2021 (IF: 0.93)

Other peer reviewed journals

[A.4] M. Trnka and T. Cerny, “Authentication and authorization rules sharing for internet of things”, *Software Networking*, vol. 2018, no. 1, pp. 35–52, 2018, ISSN: 2445-9739. DOI: 10.13052/jsn2445-9739.2017.003 (Citations: 3)
Selected citation: [B.18]

In proceedings indexed in ISI

[A.5] M. Trnka and T. Cerny, “Identity management of devices in internet of things environment”, in *2016 6th International Conference on IT Convergence and Security (ICITCS)*, 2016, pp. 1–4. DOI: 10.1109/ICITCS.2016.7740343 (Citations: 15)
Selected citation: [B.7] [B.14] [B.15]

[A.6] M. Trnka, M. Tomasek, and T. Cerny, “Context-aware security using internet of things devices”, in *Information Science and Applications 2017*, 2017, pp. 706–713, ISBN: 978-981-10-4154-9. DOI: 10.1007/978-981-10-4154-9_81 (Citations: 3)

[A.7] M. Trnka, F. Rysavy, T. Cerny, and N. Stickney, “Using wi-fi enabled internet of things devices for context-aware authentication”, in *Information Science and Applications 2018*, 2019, pp. 635–642, ISBN: 978-981-13-1056-0. DOI: 10.1007/978-981-13-1056-0_62 (Citations: 2)

[A.8] T. Cerny, M. Trnka, and M. J. Donahoo, “Towards shared security through distributed separation of concerns”, in *Proceedings of the International Conference on Research in Adaptive and Convergent Systems*, 2016, pp. 169–172, ISBN: 9781450344555. DOI: 10.1145/2987386.2987394

Other proceedings

[A.9] M. Trnka and T. Cerny, “On security level usage in context-aware role-based access control”, in *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, 2016, pp. 1192–1195, ISBN: 9781450337397. DOI: 10.1145/2851613.2851664 (CORE ranking: B, Citations: 23)
Selected citations: [B.2], [B.9], [B.10], [B.12], [B.13], [B.19], [B.20]

[A.10] M. Trnka and T. Cerny, “Context-aware role-based access control using security levels”, in *Proceedings of the 2015 Conference on Research in Adaptive and Convergent Systems*, 2015, pp. 280–284, ISBN: 9781450337380. DOI: 10.1145/2811411.2811498 (Citations: 3)

[A.11] M. Trnka, J. Svacina, T. Cerny, and E. Song, “Aspect oriented context-aware and event-driven data processing for internet of things”, in *Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems*, 2018, pp. 319–323, ISBN: 9781450358859. DOI: 10.1145/3264746.3264761

[A.12] (Accepted) M. Bures, B. S. Ahmed, V. Rechtberger, M. Klima, M. Trnka, M. Jaros, X. Bellekens, D. Almog, and P. Herout, "Patriot: Iot automated interoperability and integration testing framework", in *2021 IEEE 14th International Conference on Software Testing, Validation and Verification (ICST)*, 2021 (CORE ranking: A)

Unrelated Publications

Other peer reviewed journals

[A.13] T. Cerny, M. J. Donahoo, and M. Trnka, "Contextual understanding of microservice architecture: Current and future directions", *SIGAPP Appl. Comput. Rev.*, vol. 17, no. 4, pp. 29–45, 2018, ISSN: 1559-6915. DOI: 10.1145/3183628.3183631 (Citations: 76, Citations with IF: 28)

In proceedings indexed in ISI

[A.14] J. Sebek, M. Trnka, and T. Cerny, "On aspect-oriented programming in adaptive user interfaces", in *2015 2nd International Conference on Information Science and Security (ICISS)*, 2015, pp. 1–5. DOI: 10.1109/ICISSEC.2015.7371024 (Citations: 3, Citations with IF: 1)

Selected Citations

Below I list selected citations from articles in journals with Impact Factor and are referencing only publications related to the dissertation topic.

[B.1] M. Barbareschi, A. D. Benedictis, E. L. Montagna, A. Mazzeo, and N. Mazzocca, "A puf-based mutual authentication scheme for cloud-edges iot systems", *Future Generation Computer Systems*, vol. 101, pp. 246–261, 2019, ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2019.06.012 (IF: 6.13)

[B.2] A. Kayes, W. Rahayu, T. Dillon, E. Chang, and J. Han, "Context-aware access control with imprecise context characterization for cloud-based data resources", *Future Generation Computer Systems*, vol. 93, pp. 237–255, 2019, ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2018.10.036 (IF: 6.13)

[B.3] J.-P. A. Yaacoub, M. Noura, H. N. Noura, O. Salman, E. Yaacoub, R. Couturier, and A. Chehab, "Securing internet of medical things systems: Limitations, issues and recommendations", *Future Generation Computer Systems*, vol. 105, pp. 581–606, 2020, ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2019.12.028 (IF: 6.13)

[B.4] M. Mahbub, "Progressive researches on iot security: An exhaustive analysis from the perspective of protocols, vulnerabilities, and preemptive architectonics", *Journal of Network and Computer Applications*, p. 102 761, 2020, ISSN: 1084-8045. DOI: `https://doi.org/10.1016/j.jnca.2020.102761` (IF: 5.57)

[B.5] G. Fortino, F. Messina, D. Rosaci, and G. M. L. Sarne, "Resiot: An iot social framework resilient to malicious activities", *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 5, pp. 1263–1278, 2020. DOI: `10.1109/JAS.2020.1003330` (IF: 5.12)

[B.6] M. Akil, L. Islami, S. Fischer-Hübner, L. A. Martucci, and A. Zuccato, "Privacy-preserving identifiers for iot: A systematic literature review", *IEEE Access*, vol. 8, pp. 168 470–168 485, 2020 (IF: 4.64)

[B.7] R. Sardar and T. Anees, "Web of things: Security challenges and mechanisms", *IEEE Access*, pp. 1–1, 2021. DOI: `10.1109/ACCESS.2021.3057655` (IF: 4.64)

[B.8] R. H. Aswathy and N. Malarvizhi, "A design of lightweight ecc based cryptographic algorithm coupled with linear congruential method for resource constraint area in iot", *Journal of Ambient Intelligence and Humanized Computing*, Jan. 2021, ISSN: 1868-5145. DOI: `10.1007/s12652-020-02788-0` (IF: 4.59)

[B.9] A. S. M. Kayes, R. Kalaria, I. H. Sarker, M. S. Islam, P. A. Watters, A. Ng, M. Hammoudeh, S. Badsha, and I. Kumara, "A survey of context-aware access control mechanisms for cloud and fog networks: Taxonomy and open research issues", *Sensors*, vol. 20, no. 9, p. 2464, Apr. 2020, ISSN: 1424-8220. DOI: `10.3390/s20092464` (IF: 3.28)

[B.10] J. Ji, G. Wu, J. Shuai, Z. Zhang, Z. Wang, and Y. Ren, "Heuristic approaches for enhancing the privacy of the leader in iot networks", *Sensors*, vol. 19, no. 18, p. 3886, Sep. 2019, ISSN: 1424-8220. DOI: `10.3390/s19183886` (IF: 3.28)

[B.11] P. Nespoli, M. Zago, A. Huertas Celdrán, M. Gil Pérez, F. Gómez Mármol, and F. J. García Clemente, "Palot: Profiling and authenticating users leveraging internet of things", *Sensors*, vol. 19, no. 12, p. 2832, Jun. 2019, ISSN: 1424-8220. DOI: `10.3390/s19122832` (IF: 3.28)

[B.12] Z.-Y. Wu, "A secure and efficient digital-data-sharing system for cloud environments", *Sensors*, vol. 19, no. 12, p. 2817, Jun. 2019, ISSN: 1424-8220. DOI: `10.3390/s19122817` (IF: 3.28)

[B.13] X. C. Yin, Z. G. Liu, B. Ndibanje, L. Nkenyereye, and S. M. Riazul Islam, "An iot-based anonymous function for security and privacy in healthcare sensor networks", *Sensors*, vol. 19, no. 14, p. 3146, Jul. 2019, ISSN: 1424-8220. DOI: `10.3390/s19143146` (IF: 3.28)

[B.14] A. Čolaković and M. Hadžialić, "Internet of things (iot): A review of enabling technologies, challenges, and open research issues", *Computer Networks*, vol. 144, pp. 17–39, 2018, ISSN: 1389-1286. DOI: `https://doi.org/10.1016/j.comnet.2018.07.017` (IF: 3.11)

[B.15] P. R. Sousa, J. S. Resende, R. Martins, and L. Antunes, "The case for blockchain in IoT identity management", *Journal of Enterprise Information Management*, ISSN: 1741-0398. DOI: `{10.1108/JEIM-07-2018-0148}` (IF: 2.66)

[B.16] L. Guo, J. Wang, and W.-C. Yau, "Efficient hierarchical identity-based encryption system for internet of things infrastructure", *Symmetry*, vol. 11, no. 7, p. 913, Jul. 2019, ISSN: 2073-8994. DOI: `10.3390/sym11070913` (IF: 2.65)

[B.17] M. Tahir, M. Sardaraz, S. Muhammad, and M. Saud Khan, "A lightweight authentication and authorization framework for blockchain-enabled iot network in health-informatics", *Sustainability*, vol. 12, no. 17, p. 6960, Aug. 2020, ISSN: 2071-1050. DOI: `10.3390/su12176960` (IF: 2.59)

[B.18] S. Pešić, M. Ivanović, M. Radovanović, and C. Bădică, "Caavi-rics model for observing the security of distributed iot and edge computing systems", *Simulation Modelling Practice and Theory*, p. 102 125, 2020, ISSN: 1569-190X. DOI: `https://doi.org/10.1016/j.simpat.2020.102125` (IF: 2.22)

[B.19] A. S. M. Kayes, W. Rahayu, and T. Dillon, "Critical situation management utilizing IoT-based data resources through dynamic contextual role modeling and activation", *Computing*, vol. 101, no. 7, SI, 743–772, Jul. 2019, ISSN: 0010-485X. DOI: `10.1007/s00607-018-0654-1` (IF: 2.04)

[B.20] A. S. M. Kayes, J. Han, W. Rahayu, T. Dillon, M. S. Islam, and A. Colman, "A Policy Model and Framework for Context-Aware Access Control to Information Resources", *The Computer Journal*, vol. 62, no. 5, pp. 670–705, Jul. 2018, ISSN: 0010-4620. DOI: `10.1093/comjnl/bxy065` (IF: 1.08)

[B.21] Z. Houhamdi and B. Athamena, "Identity identification and management in the internet of things", *The International Arab Journal of Information Technology*, vol. 17, pp. 645–654, Jul. 2020. DOI: `10.34028/iajit/17/4A/9` (IF: 0.467)