BlueSky: Towards Convergence of Zero Trust Principles and Score-Based Authorization for IoT Enabled Smart Systems

Safwa Ameer
Institute for Cyber Security and NSF C-SPECC Center, Department of Computer Science, The University of Texas at San Antonio
San Antonio, TX, USA
safwa.ameer@gmail.com

Maanak Gupta
Tennessee Technological University
Cookeville, TN, USA
mgupta@tntech.edu

Smriti Bhatt
Purdue University
West Lafayette, IN, USA
smbhatt@purdue.edu

Ravi Sandhu
Institute for Cyber Security and NSF C-SPECC Center, Department of Computer Science, The University of Texas at San Antonio
San Antonio, Texas, USA
ravi.sandhu@utsa.edu

Abstract

Zero trust (ZT) is the term for an evolving set of cybersecurity paradigms that move defenses from static, network-based perimeters to focus on users, assets, and resources. It assumes no implicit trust is granted to assets or user accounts based solely on their physical or network location. We have billions of devices in IoT ecosystems connected to enable smart environments, and these devices are scattered around different locations, sometimes multiple cities or even multiple countries. Moreover, the deployment of resource-constrained devices motivates the integration of IoT and cloud services. This adoption of a plethora of technologies expands the attack surface and positions the IoT ecosystem as a target for many potential security threats. This complexity has outstripped legacy perimeter-based security methods as there is no single, easily identified perimeter for different use cases in IoT. Hence, we believe that the need arises to incorporate ZT guiding principles in workflows, systems design, and operations that can be used to improve the security posture of IoT applications. This paper motivates the need to implement ZT principles when developing access control models for smart IoT systems. It first provides a structured mapping between the ZT basic tenets and the PEI framework when designing and implementing a ZT authorization system. It proposes the ZT authorization requirements framework (ZT-ARF), which provides a structured approach to authorization policy models in ZT systems. Moreover, it analyzes the requirements of access control models in IoT within the proposed ZT-ARF and presents the vision and need for a ZT score-based authorization framework (ZT-SAF) that is capable of maintaining the access control requirements for ZT IoT connected systems.

SACMAT '22, June 8–10, 2022, New York, NY, USA

ACM ISBN 978-1-4503-9357-7/22/06…$15.00
https://doi.org/10.1145/3532105.3535020

CCS Concepts

• **Security and privacy** → *Information flow control.*

Keywords

IoT, Cyber physical systems, Access control, Authorization, Zero Trust, Score based

ACM Reference Format:
Safwa Ameer, Maanak Gupta, Smriti Bhatt, and Ravi Sandhu. 2022. BlueSky: Towards Convergence of Zero Trust Principles and Score-Based Authorization for IoT Enabled Smart Systems. In *Proceedings of the 27th ACM Symposium on Access Control Models and Technologies (SACMAT) (SACMAT '22), June 8–10, 2022, New York, NY, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3532105.3535020

1 Introduction

As we move toward an Internet of things (IoT) enabled world, we have billions of objects capable of sensing, communicating, and sharing information, all connected to each other through public or private networks. Data from these interconnected objects (smart things) is regularly gathered, analyzed, and used to initiate actions, becoming a rich source of intelligence for planning, management, and decision-making [41]. By 2025, there will be more than 25 billion connected smart devices [11, 53]. In addition to computers, IoT objects include everyday objects that can be read, recognized, located, addressed through information sensing devices, and controlled via the Internet. These objects may reside in different faraway locations, all connected to communicate and share information to enable IoT environments in many application domains. Innovative smart IoT application domains include consumer applications (smart homes, elder care), organizational applications (medical and health care, vehicular communication systems), industrial applications (manufacturing, agriculture), infrastructure applications (smart cities, energy management), and military applications (Internet of Battlefield Things) [1]. The ultimate goal is to establish an autonomous smart ecosystem where everything is connected, continuously communicating, sharing information, and triggering actions. The use of Internet-connected devices and a large number of other supporting technologies (e.g., cloud computing, machine learning) in this smart vision, however, pose several security and privacy challenges

and concerns [35, 45]. Many researchers agree that authentication and authorization are critical aspects of the IoT [25, 35]. Therefore, a systematic and dynamic research approach is essential for IoT to maintain its success over the long term in terms of securing access, authorization, communication, and data flow [11]. Toward this goal, in this research, we motivate the need for integrating the recently proposed zero trust (ZT) paradigm principles and concepts in designing and developing IoT access control systems [46]. We envision developing a set of authorization models specifically designed for integrated ZT IoT systems.

2 Motivation

Zero trust (ZT) paradigm provides a collection of concepts and ideas designed to minimize uncertainty in enforcing accurate, least privilege per-request access decisions in information systems and services in the face of a network viewed as compromised. Zero trust architecture (ZTA) is an enterprise's cybersecurity plan that utilizes ZT concepts and encompasses component relationships, workflow planning, and access policies [46]. Hence, an integrated zero trust IoT system is the network infrastructure (physical and virtual) and operational policies that are in place for an IoT system as the result of a ZTA plan.

We believe that integrating ZT concepts is crucial when developing IoT systems for the following reasons. To begin, ZT is a response to enterprise network trends that include remote users, bring your own device (BYOD), and cloud-based assets that do not fall within the enterprise's network boundaries [46], which is typical for IoT use cases. ZT security models assume that an attacker is present in the environment and that an enterprise-owned environment should not be treated differently than any other environment owned by a third party. In this new paradigm, an enterprise must assume no implicit trust, continually evaluate the risks to its assets and business functions, and then enact protections to mitigate these risks. On the other hand, IoT systems possess some characteristics [12, 41] that make them need to integrate ZT paradigms into their authentication and authorization designs. Below are some of these characteristics: **(i) Heterogeneity.** Different types of IoT devices have varying sizes and functionality and different communication and networking mechanisms or protocols. Furthermore, they are made by different vendors with diverse platforms and protocols. Adopting such a plethora of technologies in IoT enlarges the attack surface and introduces new security vulnerabilities [35, 45]. **(ii) Distributed and Remote Location.** IoT devices can be dispersed and remotely located in different locations without the owner having any physical access to these devices. This introduces many security vulnerabilities, such as insecure access to the web, backend APIs, cloud, and mobile interfaces. Access control should be enforced at each interface in this complex IoT ecosystem. However, commercial IoT frameworks fall short in doing so [45]. **(iii) Enormous scale.** In IoT ecosystems, the number of devices that need to be managed and communicate with each other is growing exponentially. The generated data is incredibly enormous. Security and management of these data are becoming a nightmare. **(iv) Autonomicity.** When deployed, IoT devices can be autonomous using technologies such as artificial intelligence and machine learning in conjunction with cloud resources. **(v) Dynamic Behavior.** The behavior of IoT devices varies depending on the user characteristics and context in which they are used. **(vi) Connectivity and Interconnectivity.** Connectivity enables network accessibility and compatibility. Interconnectivity means that in IoT, anything can be connected to anything and to the global information and communication network. This characteristic enlarges the attack surface in IoT. **(vii) Things-related services.** As a result of the IoT, things can provide services within the constraints of things, such as privacy protection and semantic consistency between physical things and their associated virtual things. **(viii) Safety.** We must ensure IoT safety. Our personal data and our physical well-being must be secured. For example, in the context of smart healthcare, IoT devices in the form of wearables might provide data about patients such as sleep patterns, activity levels, temperature, etc. Securing endpoints, networks, and data moving across all IoT elements requires the adoption of a security paradigm that will scale.

The shortcoming in applying access control policies in IoT applications leads devices and apps to be easily exploited to gain unauthorized access to devices and users' and devices' data [14, 19, 35]. The need arises for a systematic and dynamic research approach for IoT to maintain its success over the long term in securing access, authorization, communication, and data flow. Consider the use case - illustrated in Figure 1 - of a patient with an elevated heart rate. The wearable will instantly transmit the signal to the medical staff. Further, the emerging IoT ecosystem will enable remote health care providers to provide timely assistance to their patients through face-to-face communication over the Internet or prepare care plans in advance of their arrival. Additionally, the collected data can be saved on cloud-based servers. Cloud applications can later use these data to generate intelligent charts and diagrams that can be easily analyzed by health care professionals or automatically transmitted to research centers. Furthermore, advances in artificial intelligence and machine learning may create opportunities for the healthcare system to analyze collected data, identify patterns, draw conclusions, and trigger alarms. In such a complex setting, we have different resources provided by various vendors that reside in different locations (sometimes even different countries). Those resources are accessed by multiple users with varied privileges to perform different tasks. Moreover, considering the criticality of the IoT services and the sensitivity of stored information, managing security and privacy in IoT becomes even more challenging. For instance, in our healthcare use case, a compromised wearable device can send incorrect data to health care providers. We may also have a compromised doctor account trying to update sensitive patient information or a compromised health monitoring application drawing misleading conclusions or triggering false alarms. Toward a secure IoT ecosystem, we believe that integrating ZT principles and guidelines when developing and implementing IoT connected systems is critical.

The main contributions of this paper are as follows.

- We highlight the importance of considering ZT tenets when designing, enforcing, and implementing authorization models. For this purpose, we provide a structured mapping between the ZT tenets and the PEI models framework [50]. The purpose is to identify during which part of the design process we need to incorporate the ZT tenets when developing an authorization model for integrated ZT IoT systems.

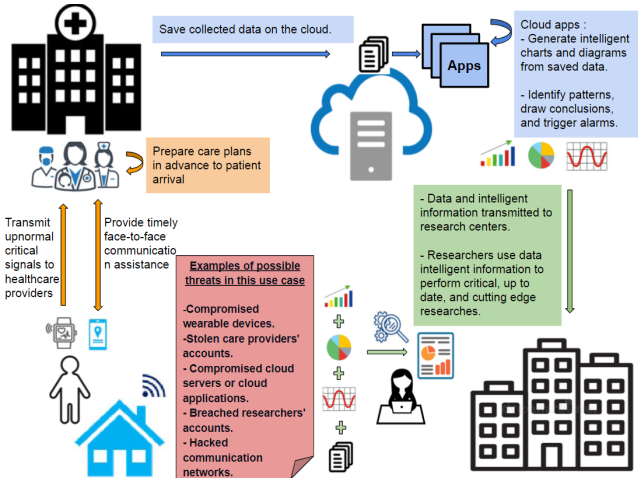

Figure 1: Healthcare Use Case

- We propose the ZT authorization requirements framework (ZT-ARF), which provides a structured view of different authorization requirements to consider when designing a ZT authorization policy model.
- We analyze access control requirements in IoT systems and accordingly specify which requirements components from our proposed ZT-ARF we need to include when designing an authorization model for integrated ZT IoT systems.
- We propose our novel framework for ZT score-based authorization access control policy model (ZT-SAF).
- We highlight future research directions and propose a plan for designing, enforcing, and implementing the proposed ZT-SAF in smart connected IoT systems.

3 Related Work and Background

3.1 Related work

Security and privacy in smart connected IoT systems present unique challenges [5, 17, 22]. Among those challenges, providing access control is considered critical by most researchers. Several approaches have been used in enforcing access control policies, including cryptographic mechanisms, capabilities, access control lists, and policy based solutions [15, 20, 54]. Some solutions are based on RBAC [21, 47] (as in [2, 7, 10, 16]). However, RBAC based models do not capture different dynamic attributes [29]. Other proposals are based on ABAC [26, 27] (as in [3, 4, 9, 11]). Both ABAC and RBAC based models do not support continuous authorization control. Some of the proposed models in the literature are built on blockchain technology [33, 34]. However, as [33] described, blockchain technology has some technical characteristics that could limit its applicability. For instance, cryptocurrency fees and processing time. Several other access control models for IoT have been proposed; the authors in [35, 43, 45] provided surveys on them. Few solutions have been proposed in the literature for IoT systems based on UCON model [37, 38, 42, 49]. However, these models cannot be adopted yet for various reasons. In [23], the model is proposed as a Device to Services (D-S) access control model. Moreover, no implementation was provided. In [30], the authors mainly focused on providing a distributed Peer-to-Peer (P2P) architecture. They did not consider using their system to grant users access to different smart things.

3.1.1 Risk-Adaptive Access Control The concept of score-based access control was introduced in [32] as risk-adaptive access control models RAdAC, but without articulation of a precise formal model. Accordingly, the authors in [28] proposed an attribute-based framework for RAdAC. Moreover, the authors in [13] proposed a framework for risk-aware role-based access control. In the literature, few risk-adaptive solutions have been proposed for IoT systems. In [31], the authors provide an enforcement architecture framework for a qualitative risk-based usage control model. However, they didn't provide a formal policy model for their framework. The authors in [6] proposed an enforcement architecture model for risk-adaptive access control in IoT. However, they neither provided a formal policy model for their architecture nor implemented their model. In [8], the authors proposed a framework that extends the role-based access control model by incorporating a risk assessment process based on the trust the system has on the users only. However, in a dynamic environment such as IoT, the trust level calculation process may need to incorporate resource status and environmental conditions in addition to user information. Moreover, as shown in [3], RBAC models fail to capture dynamic attributes of users, devices, and environment. In addition, they do not support ongoing authorization. In [18], the authors proposed a risk or trust aware authorization framework for IoT by extending an ABAC model. However, they only introduced an enforcement architecture for their framework without providing a formal policy model.

In this paper, toward zero trust IoT systems, we motivate the need for a score-based access control framework capable of continuously monitoring and re-evaluating policies to grant or revoke access to different resources in smart connected IoT systems. This framework needs to consider different dynamic characteristics in the IoT environment. Furthermore, it needs to consider different risk factors in such a smart connected context when calculating the trust score or the current risk level.

3.2 The PEI Models Framework

The PEI framework was first introduced in [50]. This framework was developed to meet modern distributed systems design challenges. It is difficult to close the policy-mechanism gap in one step in such a complex environment. Thus, the PEI framework separates design into three layers. These are the Policy (P), Enforcement (E), and Implementation (I) layers. We need formal models to express and analyze the security policy at each layer. See Figure 2 for the complete PEI framework. The top layer is the objectives layer, and it is intentionally informal and intended to sketch out high-level security and system goals. The bottom layer consists of the actual running code based on trusted computing technology. PEI consists of three interdependent inner layers with many to many relationships between them. For instance, a policy model at the P layer may have many different manifestations at the E layer. A policy model uses a formal or quasi-formal notation to formalize informal high-level objectives. It usually has rigorous mathematical foundations. While the policy models focus on the "what" aspect, the enforcement and implementation models aim to answer the "how" question at the level of system block diagrams and protocol flows.

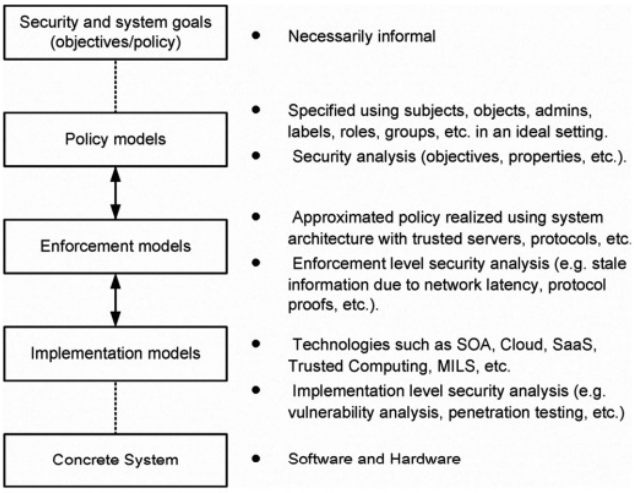

Figure 2: The PEI Models Framework [50].

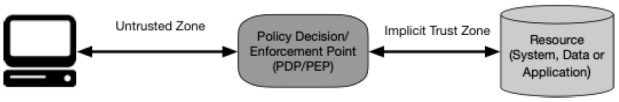

Figure 3: Zero Trust Access [46].

Finally, the implementation layer spells out detailed implementation protocols and mechanisms. The concept of a model arises at all three layers. At the enforcement layer, the term architecture is also used, while architecture and platform are also used at the implementation layer.

3.3 Zero Trust Systems

ZT focuses on authentication, authorization, and shrinking implicit trust zones to reduce uncertainties while maintaining availability and minimizing delayed authentication times. Figure 3 depicts the abstract ZT model of access [46]. It is the PDP/PEP that makes the appropriate decision regarding whether or not to allow the subject to access the resource. Towards these goals, the authors in [46] attempt to define ZT and ZTA in terms of seven tenets that should be included rather than what is excluded. These tenets are essential for obtaining a secure, trusted access control system. Furthermore, they introduce an enforcement architecture that illustrates the ZT policy engine's trust algorithm inputs as shown in Figure 4. The trust algorithm (TA) is the process used by the policy engine to ultimately grant or deny access to a resource, in other words, the authorization model. However, they did not provide a formal policy model for trust algorithms (authorization models).

4 A Structured Approach Towards Access Control in Zero Trust Systems

Access control defines who has access to what, when, and in which conditions [52]. In general, access control requires both authentication and authorization techniques. Authentication is any process by which a system verifies the identity of a user who wishes

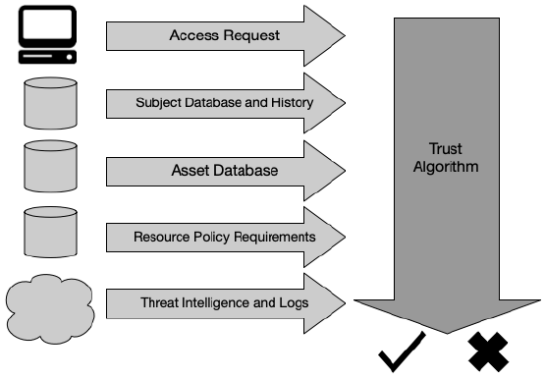

Figure 4: Trust Algorithm Input [46].

to access the system, and authorization determines what an authenticated user can or cannot do in the system.

ZT provides a collection of concepts that focus on authentication, authorization, and shrinking implicit trust zones to reduce uncertainties. In line with these goals, the authors in [46] offer a way of defining ZT and ZTA in terms of basic tenets that should be adhered to when designing and deploying a ZT system. The question remains, however, when we must consider different tenets. Particularly at which design and implementation stages should we incorporate each tenet into the access control system. As a result, in Section 4.1, we provide a structured mapping between the ZT basic tenets and the PEI framework when designing and implementing a ZT authorization system. Moreover, Section 4.2 proposes the ZT authorization requirements framework. It provides a structured approach to investigate and consider different requirements when designing a ZT authorization system. Our main focus is authorization, and authentication is outside the scope of this investigation.

4.1 Mapping Between Zero Trust Tenets and The PEI Models Framework Layers

In the following, we describe each tenet and the different design layers within the PEI models framework that need to capture each tenet in ZT authorization systems. First, since the ZT tenets are considered the ideal goals for zero trust systems, every tenet to be enforced must be addressed in the objectives layer.

Tenet 1: All data sources and computing services are considered resources. This tenet emphasizes that no trust should be granted to any data sources or services regardless of their owners or location. In addition to the objectives layer, this tenet should also be considered at the policy models layer. The policy model should take into consideration different types of resources. For instance, in some applications such as social computing systems, users or users' sessions are considered data sources. Hence, according to this tenet, they should be treated as resources. In many cases, different resources are handled differently at the policy level. A user to device authorization model may have different requirements than an authorization model that governs users to session requests or users to service requests. Accordingly, the enforcement architecture model should ensure that all data sources and computing services requests are mediated through the policy decision and enforcement points. Moreover, this tenet is also captured in the implementation

models layer since implementation models should maintain the workflows designed in the enforcement layer.

Tenet 2: All communication is secured regardless of network location. The authorization policy model should ensure that access requests are not granted based on the network location. Moreover, the enforcement architecture model should ensure that all access requests are mediated through the policy decision and enforcement points regardless of network location. Accordingly, the implementation model should maintain the workflows specified by the enforcement architecture model.

Tenet 3: Access to individual enterprise resources is granted on a per-session basis. Authorization policy models should incorporate the concept of sessions, and they should provide actors with the minimum privileges needed to complete their tasks within a particular session. Consequently, the enforcement architecture and implementation models should incorporate the network components and software technologies to enable the concept of sessions.

Tenet 4: Access to resources is determined by dynamic policy—including the observable state of client identity, application/service, and the requesting asset—and may include other behavioral and environmental attributes. When deciding on an access request, policy models should be capable enough to consider different clients, resources, assets, operations, services, and environmental characteristics. Furthermore, policy models should monitor the behavior of different components in the system to measure these components' deviations from observed normal usage patterns. Moreover, enforcement architectures and implementation models should incorporate components that sense and monitor those attributes and send it as input to policy decision points.

Tenet 5: The enterprise monitors and measures the integrity and security posture of all owned and associated assets. This tenet asserts that no asset is inherently trusted regardless of its owner. Hence, the access control policy model should consider the security status of different assets involved within a request before granting access. Accordingly, the enforcement architecture and the implementation models should include components that evaluate and measure the security posture of different assets.

Tenet 6: All resource authentication and authorization are dynamic and strictly enforced before access is allowed. From an authorization point of view, this tenet emphasizes that access control authorization policy models in ZT systems should reevaluate trust in ongoing communication dynamically and continuously. Hence, the enforcement architecture model and the implementation model of the authorization system should have access management components in place that can continuously evaluate and reauthorize ongoing access.

Tenet 7: The enterprise collects as much information as possible about the current state of assets, network infrastructure and communications and uses it to improve its security posture. Accordingly, authorization policy models should continuously keep track of the current state of assets and the network and communication information and use this information to update and improve policy creation and enforcement. To achieve this, authorization enforcement architecture and implementation models should include advanced components capable of monitoring different asset statuses and network and communication information and use this information to generate updated policies.

We can conclude that when designing a zero trust authorization system, it is critical to consider the zero trust tenets that we would like to implement at the policy, enforcement, and implementation models layers in the PEI framework. As a result, since the concrete system layer represents the hardware and software used to execute the processes and workflows outlined on the other layers, all tenets need to be captured at the concrete system layer.

4.2 The Zero Trust Authorization Requirements Framework (ZT-ARF)

As mentioned in the ZT NIST document [46], although ZT basic tenets are the ideal goals, not all of them may be implemented in their purest form in every system. There are no minimum requirements in terms of tenets or principles, and instead, It is left to be decided on by the system architect according to different application domains, business needs, challenges, and requirements. Hence, different enterprises may choose to fully or partially incorporate some tenets while neglecting others.

This section proposes the zero trust authorization requirements framework (ZT-ARF) based on the ZT tenets. The motivation is to provide authorization policy models designers with a structured view of different authorization requirements to consider when designing a zero trust authorization policy model. Figure 5 depicts the framework components. In categorizing these requirements, we provide a family of seven requirements components: actor characteristics, target characteristics, action characteristics, action-target characteristics, context characteristics, usage check, and behavioral check. Moreover, here we describe a structured approach for defining packages of requirement components, where each package may be appropriate to different application domains, threat environments, and market segments. Each component can optionally be selected for inclusion into a package with one exception; the actor characteristics component is required as part of all packages. In general, the more components that are included in an authorization model, the closer that model comes to ZT ultimate goals. Here, we should mention that this framework addresses the requirements of operational authorization policy models. Administrative authorization policy models requirements are outside our scope.

Actor Characteristics. The actor is the system entity that initiates the access request. Actors may include users, subjects [27], system services, applications, and devices. A user is a human being who requests access to the system. Devices may be the main actors in some application domains. For example, in IoT, we have different devices that exchange data and trigger different actions. Moreover, in some cases, we may have different applications that trigger different access requests in the system. System service can also act as an actor, such as an automated data request service. Some systems allow different actors to create sessions or subjects to perform some operations on the system. Sessions usually inherit part or all the characteristics of their actors. Actors can be operators or administrators. Operators have access to some or all usage operations and services offered by the system. Administrators have access to some or all administrative operations and services. Actor characteristics (attributes) represent the actor's prosperities, such as ID, clearance, roles, gender, trust level, location, etc. In this framework, we consider the characteristics of the assets used by actors to initiate different actions as part of actor characteristics.

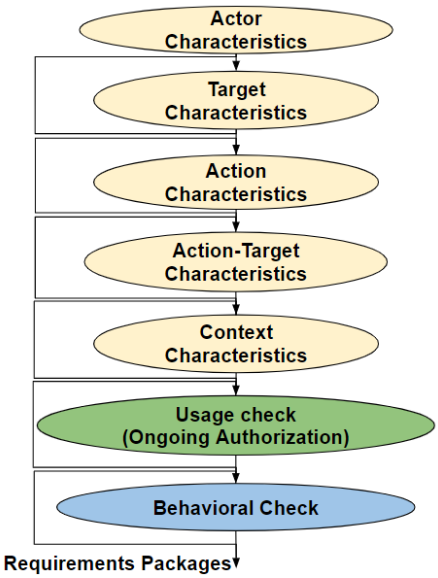

Figure 5: The Zero Trust Authorization Requirements Framework (ZT-ARF)

Target Characteristics. Targets are the resources that need to be accessed in the request. Targets can include information containers (e.g., files or directories in an operating system and columns, rows, tables, and views within a database management system). Targets may also include applications and devices which contain operations to be performed and data to be accessed. Moreover, in some application domains such as social computing, targets may include users and sessions. For instance, Alex may invite Anne as a friend or request a chat with Anne. Here Anne is the target user while Alex is the acting user. In such cases, although the action is made to a target user, it is the target user's session(s) that receives the action. Here it is Anne's session that receives Alex's invitation. Targets are associated with characteristics, such as target location, target owner, target security posture, and so on.

Action characteristics. An action is an executable image of a program, which executes some function for the actor after invocation. For example, operations. Actions may be associated with different characteristics. For instance, in the environment of smart homes, we may want to characterize dangerous kitchen operations or kids-friendly operations. Fine grained authorization system should be capable of capturing different action characteristics.

Action-Target characteristics. Actions are usually associated with different targets, and an approval to perform an action on a specific target gives us the action-target combination. For instance, the concept of permission in RBAC systems [48], and the concept of activities in activity-centric access control systems [24, 39, 40]. In some cases, the action-target combination concept has characteristics to be captured, for instance, the activity level of danger.

Context characteristics. This includes environment characteristics, system characteristics, threats, and log information. Environment characteristics describe the environmental conditions such as time, weather, global alarm hazards, etc. System characteristics describe the application system condition, for instance, the operating system version, the security status of different system components, etc. Threats and log information include the log information of different assets, network infrastructure, and communication and the feeds about threats operating on the Internet and the system.

Usage check. As discussed in Section 4.1, tenet 6 emphasizes the need for a dynamic, ongoing authorization in ZT systems. This requirements component represents the continuity of an access decision, and it implies that the authorization system enforces the security policy before the access execution, during the execution, and afterward. If access requirements or attributes change during access and the security policy is no longer satisfied, the authorization system should revoke the granted access and stop usage or perform other actions as specified by the policy specification.

Behavioral check. Tenet 4 in the ZT basic tenets highlights the need for closely observing the behaviors of different components in the system to capture any abnormal behavior. As a result, this requirements component addresses the need to incorporate behavioral checks in the authorization process.

In the five top requirements components, we distinguish between two types of characteristics: static and dynamic. All characteristics indeed may change over a long time. However, some attributes are "relatively" static, as they evaluate to the same values over a long period of time, such as the user set of roles, device owner, etc. On the other hand, dynamic attributes constantly change due to various circumstances, such as time of the day, actor location, etc.

4.3 ZT-ARF Analysis

In this section, we go through the ZT basic tenets and demonstrate that the ZT-ARF proposed in Section 4.2 above is capable enough to fulfill the authorization requirements proposed by the ZT basic tenets. Tenet 1 requires the authorization policy model to consider different types of resources. Our proposed authorization requirements framework is flexible enough to capture different types of resources, whether those resources are actors or targets. Tenet 2 emphasizes that network location alone does not imply trust. ZT-ARF is comprehensive enough to capture and use different characteristics other than network location when deciding on an access request. It can capture actor, target, action, action-target, and context static and dynamic characteristics. Tenet 3 requires incorporating the concept of sessions in authorization policy models. In ZT-ARF, the first requirements component is flexible enough to include sessions as actors and consider their characteristics in the authorization process. Tenet 4 addresses the need for dynamic policies when deciding on access requests. Actor characteristics, target characteristics, behavioral checks, and context characteristics requirements components in ZT-ARF enable the designing of dynamic access policies. Tenet 5 asserts that the access control model should consider the security status of different assets involved within a request before granting access. In a typical access request, involved assets will include the requesting asset, the requested resource asset, and the involved intermediary system assets (for example, the involved PEP/PDPs). The security status of these assets is respectively captured through actor characteristics, target characteristics, and the context characteristics component. Tenet 6 highlights the need for continually reevaluating trust in ongoing communication. The usage check requirements component in ZT-ARF maintains this tenet. Finally, Tenet 7 identifies the importance of continuously

monitoring and collecting information about different assets, infrastructure, and communications in the system and utilizing this information to improve policy creation and enforcement. The ZT-ARF is comprehensive enough to capture this information through the actor's, target's, and context's characteristics requirements components and continuously monitor the access through the usage check requirement component. This paper focuses on operational access control authorization models. Policy creation and management tasks are considered part of administrative access control, hence outside the scope of this framework.

5 Toward Zero Trust Authorization In IoT Ecosystem

IoT environments, where multiple parties collaboratively create, share, manage and protect digital content and other resources, require a sophisticated access control system to handle such complex interdependent activities [2, 36]. In this section, we describe the authorization requirements in IoT systems. Furthermore, we map these requirements to our proposed zero trust authorization requirements framework. The main goal is to decide which requirements components within the ZT-ARF we need to include when designing zero trust authorization models for IoT systems.

5.1 Authorization Requirements in Zero Trust IoT Connected Systems

Ouaddah et al. [35] have provided a survey that identifies the main challenges of IoT access control models. Accordingly, some authors (e.g. [2, 12, 19]) introduced criteria and requirements for access control models in different IoT systems. Based on these researches and based on the IoT characteristics explained in Section 2, we introduce the following specifications for authorization models in IoT systems. These requirements are mainly considered while designing new access control models and/or adapting existing access control models for IoT systems.

(i) Dynamic Authorization. The authorizations within such a highly dynamic environment vary based on the components and the context in which they are used. Hence, the model should be dynamic so as to capture actor, environment, and target contextual information. For example, in a smart home domain, the access rights will change based on the type of users (parents, kids, workers, etc.), the devices they use, the targets they are accessing (oven, TV, etc.), and the context characteristics (time, weather, etc.). Moreover, the model should be dynamic enough to enable continuous verification for access control authorized policies to facilitate ongoing controls for relatively long-lived operations or immediate revocation. For instance, in the same smart home example, if the time is currently 5:50 pm and kids are watching TV, they are only allowed to watch it till 6:00 pm. The system should enable continuous authorization to revoke the TV access at 6:00 pm. **(ii) Fine-grained Authorization.** As illustrated in [44], there is a risk asymmetry between operations or actions in the same smart device. For instance, in smart home domain, "oven.on" is a potential fire hazard, "oven.off" is potentially uncooked food, "mic.on" is a privacy risk, "mic.off" might only disable certain voice-assistant functionality like Amazon Alexa, etc. Hence, the authorization model should be fine grained enough so a subset of the functionality of a device can be authorized rather than all-or-nothing access to the device. **(iii) Suitable for constrained smart devices.** Smart things are usually limited in terms of computational power and storage. Furthermore, a generic interoperability standard among IoT devices is still missing. Accordingly, the model should not require extensive computation or communication by those constrained devices. **(iv) Compliance.** IoT currently lacks access control standards. For smart communities to be successful and sustainable, it is necessary to develop new standards and models of access control that comply with those standards. **(v) Formal policy definition.** The model should have a formal definition so that the intended behavior is precise and rigorous. **(vi) Scalability.** Access control models and mechanisms must be scalable to incorporate authorizations associated with a small or large number of devices, users, and other entities in the IoT system. **(vii) Privacy-Preserving.** A large amount of data is continuously collected and shared across multiple components in smart IoT connected systems. A privacy-preserving access control model should enable a user-centric privacy approach where users own their data and information and can decide how to share it only when required.

5.2 Continuous Ongoing Authorization for IoT Systems

In IoT systems, different components (users, devices, services, and apps) usually communicate within the scope of some context, such as operating the lawnmower, watching TV, turning on the oven, etc. The context of these interactions is synchronous, often interconnected, and has temporal continuity. This requires contextual awareness of policies and the capability to manage the entire life cycle of multiple concurrent sessions relating to authorization and usage of resources. UCON [38] supports such contextual aware ongoing authorization. However, UCON's ongoing authorization monitoring is limited to the duration of an active authorization. This neither provides continuity of communication during collaborative, multi-modal, or behavioral authentication nor provides safety during access revocation [18]. For instance, denying authorization at a given instance may not necessitate abortion of the authorization process but instead change the monitored session state towards a further iteration, to trigger additional authentication and authorization checks to obtain or maintain access.

5.3 Score-Based Authorization for IoT Systems

As mentioned in the ZT NIST document [46], one of the major characteristics that can be used to differentiate trust algorithms (authorization models) is how the input factors are evaluated to decide on access requests. Here they differentiate between two types of models: criteria-based and score-based. A criteria-based authorization model assumes certain qualifications (conditions, characteristics, etc) must be met before access to a resource (e.g., read/write) can be granted. Access is granted or action applied to a resource only if all the criteria are met. RBAC[47, 47] , UCON [47], ACON [39], and ABAC [26, 27] are all considered criteria-based authorization models. For instance, in RBAC, if a user u_i requests to execute permission p_j, he will be granted access only if he is a member of a role r_k, and the requested permission p_j is assigned to the role r_k. On the other hand, a score-based model computes a confidence level (score) for the requested access. As long as the score exceeds the threshold value configured for the resource, access to the resource is granted or the action performed. Otherwise, the request

is declined, or access privileges are reduced. Score-based authorization models are more dynamic than criteria-based models since the score provides a current confidence level for the requesting actor and adjusts to changing factors more quickly than static policies modified by human administrators [46]. The dynamism of communication between people, connected devices, data, utility, and the changing nature of the system and environment characteristics in a smart IoT connected system requires that actors' rights and access requirements change accordingly. In addition, many consumer IoT devices lack certifications and are not rigorously tested for security controls. Therefore, they are at risk of unknown or unpredictable security or privacy threats. Furthermore, many of the inputs from the sensors are subjective and probabilistic rather than absolute. Therefore, it is imperative that authorization considers the confidence level (score) of different access requests and that their policies can accommodate subjective information and uncertainty.

To develop a ZT authorization system for an IoT application domain, we need to include the following components from the proposed ZT-ARF. Actor characteristics, target characteristics, context characteristics, and usage check requirements components to build a dynamic authorization model. Action and action-target characteristics components are critical in maintaining a fine-grained authorization model. While the behavioral check requirements component provides more dynamic authorization models capable of capturing deviations from normal behaviors, it requires sophisticated policy and enforcement models. For instance, it may require incorporating machine learning and AI technologies. Hence, we believe that including the behavioral check requirements components depends on the specific IoT application domain. Since it requires a trade-off between the sensitivity of the resources and data, the business needs on the one hand, and the cost and acceptable level of complexity on the other hand. Moreover, a score-based authorization model is more dynamic in capturing IoT systems requirements than criteria based authorization model. From the above, to develop a ZT IoT system, the need arises for a contextual aware access control model capable of: (i) incorporating actor, targets, action, action-target, and context characteristics' (ii) continuously performing ongoing authorization, and (iii) dynamically deciding on access requests based on calculated score (confidence level) rather than on static access control policies.

6 Zero Trust Score-based authorization framework

This section introduces a preliminary framework for score-based authorization in terms of basic components and their interactions. Subsequently, specific formal models, policy languages, and enforcement architectures can be developed. We believe that such an abstract framework is needed to make progress in this area similar to those that underlie the successful practice of DAC, MAC, and RBAC models. Figure 6 illustrates the framework. In this framework, access is determined based on a predefined access policy and accepted trust level (score), not just a proper comparison of attributes. Here, score metrics are computed for different entities involved in the access request, and a threshold is calculated for different resources. The access decision enforcement engine (ADE) considers the calculated scores, resource threshold, and predefined

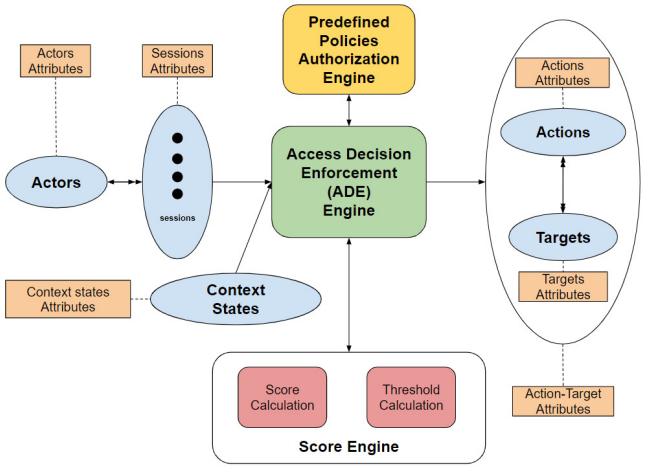

Figure 6: Zero Trust Score-based authorization framework (ZT-SAF)

policies when deciding on access. In the following, we describe the framework components.

6.1 Basic Sets and Components

Actors. This set represents the system entities that request access or initiate action. Depending on the application domain, actors may include users, devices, apps, and services. **Sessions**. The framework allows actors to create sessions during which they can perform some actions on the system. Only the users who initiate the session have the authority to terminate it. The session concept allows actors to activate the least privileges required to perform an action during a specific session. **Context states**. It is a set of states. Each state represents a picture of the context that we want to describe at a given time instant. Different states represent different time instants, such as current, yesterday, etc. Context includes environment context, system context, and threats and logs information. **Targets**. This set represents the set of resources on which actions are being performed. They are entities containing or receiving information. The targets set may include information containers, applications, and devices depending on the application domain. It may also include users and sessions (e.g., in the social computing domain). **Actions**. An action is an executable image of a program or an activity performed on a target resource following manufacturer specifications. This set may include operations, services, and activities. **Action-target**. An action-target pair represents an approval to perform an action on one target. **Attributes**. Actors, sessions, context states, targets, actions, and the different pairs of actions and targets have characteristics used in the authorization decision and expressed as their attributes. **Predefined policies authorization engine**. The task of this engine is to evaluate each access request according to a predefined authorization policy. The authorization policy should be defined using a formal policy definition language. When deciding on an access request, the authorization policy should consider different characteristics (attributes) of the components involved in the request. This engine may implement different authorization policy models depending on the application domain needs. The authorization model designers need to consider the ZT-ARF proposed in Section 4.2 when deciding on the policy model that needs

to be implemented by this engine. For instance, as we discussed in Section 5.1, in our study case, which is IoT systems, this engine should implement an ongoing authorization policy model, which in addition to different attributes values, needs to incorporate the continuous ongoing authorization requirement. **Score engine**. This engine contains two main components: the score calculation function and the threshold calculation function. It receives the access request from the ADE engine as an input and returns the access request score and the resource threshold value as an output. **Score calculation**. This function receives the access request and returns the calculated score. The score is described as a real time, measured determination of trust granted to the input access. This function implements a score calculation algorithm to calculate different access requests' scores. **Threshold calculation**. This function receives the access request as an input and returns the calculated resource threshold as an output. The threshold is described as a real-time, measured determination of acceptable trust level for the requested resource at the current instant of time. This function implements a threshold calculation algorithm to calculate different requested resources' thresholds. **Access decision enforcement (ADE) engine**. This engine decides on different access requests. It executes the following steps: (i) Receives the actor request. (ii) Sends the request to the score engine and the predefined policies authorization engine. (iii) Receives the following: the authorization engine's output, the calculated access request score, and the calculated resource's threshold. (iv) Decides whether to grant the actor's access request or not, according to its implemented algorithm.

6.2 Framework Algorithms

Score Calculation Algorithm. There could be different types of score calculation algorithms depending on the application system. It may be probabilistic, heuristic, or a simple mathematical function. Different factors affect the score calculation algorithm result. These may include the attribute values of the context and involved actor, session, target, action, action-target. We may need to include some system's historical information in heuristics algorithms.

Threshold Calculation algorithm. Here, we may have different types of algorithms. They may be probabilistic, heuristic, or simple mathematical function algorithms. Different factors affect the calculated threshold. These factors may include the requested target, action, and action-target pair attributes. They may include the context attributes values of the system. In the heuristics algorithm, these factors may include the system's historical information.

ADE Engine Algorithm. Depending on the system requirements and the acceptable levels of risk, different systems may implement different ADE algorithms. For example, the ADE engine may grant access if the output of the authorization engine is true and the calculated score of the access request is greater than the calculated threshold value. However, there may be other scenarios. For instance, the system may want to request more authentication and verification information to grant access if the output of the authorization engine is true and the calculated score of the access request is less than the calculated threshold value.

7 Future Research Agenda

This section describes future research directions to mature Score-based ongoing authorization models in ZT IoT systems successfully. In the following, we highlight some open research directions within the PEI framework recommended phases. As illustrated in Section 3.2, the PEI framework separates modern distributed systems design tasks into three layers (phases). These are the Policy (P), Enforcement (E), and Implementation (I) layers. We need formal models to express and analyze the security policy at each layer.

Authorization Operational Policy Models and Extensions. Formally defined, abstract mathematically based models similar to RBAC96 [51] and ABAC [27] are needed so that there is a precise and rigorous specification for the intended behavior. There is a need to develop a meta-model that can provide the foundation for designing extensible and adaptable access control models applicable to various IoT domains. These models should illustrate the ZT-SAF's core components, interactions, and functions. Moreover, they must address the principles of next generation access control [12]. The authorization engine should incorporate different components' characteristics when deciding on an authorization request. Moreover, it should continuously monitor the access requests for ongoing authorization. Actors, targets, and actions can be divided into several detailed components with different perspectives.

Framework Algorithms. Different score calculation, threshold calculation, and ADE engine algorithms should be developed for different application domains and business needs.

Administrative Policy Models. Administrative access control models control the access of admin (administrative) users to resources and entities. These models specify who can access the following capabilities: (i) Create actors, targets, actions, and context states. (ii) Define attributes. (iii) Create and set authorization engine policies. (iv) Modify and update the framework algorithms.

Policy Language and Constraints. It is essential to develop languages to express policies specified by the operational and administrative models. These policies must be abstract, flexible, and extensible to be applied in different IoT application domains.

Enforcement Architectures. As we mentioned in Section 3.2, while the policy models answer the "what" question, the enforcement architecture models aim to answer the "how" question at the level of system block diagrams. These models identify different components and workflows to enable the deployment of the operational and administrative formal models. Moreover, to facilitate cross enterprises, cross-domain interaction among smart connected things, federated and collaborative architectures need to be developed.

Implementation Models. Detailed models defining technical workflow mechanisms and communication protocols are required to implement operational and administrative enforcement architectures. Different implementation models can be developed depending on the platforms and underlying technologies. Moreover, security analysis investigation studies are needed to define the proposed models' security and performance challenges and limitations.

Behaviorally Aware Models. After developing mature models within the ZT-SAF, the next step would be to develop a ZT score-based behaviorally aware authorization framework. Such a framework will capture the behavioral check requirements component in the ZT-ARF described in Section 4.2. A behavioral aware authorization framework should utilize history information to detect an attacker using subverted credentials to access information in an atypical pattern of what the system sees for the given actor.

AI and Data Driven Deployment. With AI, security defenses can be automated. Similar research is needed to develop AI and

data-driven systems based on system logs and other data to automatically update the ADE engine, policy engine, score calculation, and threshold calculation algorithms.

Applications domains in IoT. The developed models for the proposed ZT-SAF must be extensible and adaptive to different IoT application domains. Once a core formal policy model is developed, this should be extended to different smart IoT systems. Following the PEI models framework in our research agenda enables us to develop interdependent models for different PEI layers with many to many relationships. For instance, a policy model at the policy layer may have many different manifestations at the enforcement layer. Moreover, an enforcement model at the enforcement layer may have many different implementations depending on the underlying technology. This feature allows us to extend and enforce different policy models in different application domains.

8 Conclusion

This paper highlights the need to integrate zero trust (ZT) concepts in IoT systems. We first analyzed the ZT tenets with respect to the PEI models framework. Accordingly, we proposed the ZT authorization requirement framework (ZT-ARF), which provides a structured approach to develop authorization models for ZT systems. Furthermore, we discussed the authorization requirements in IoT systems to determine which components we need to include from our proposed ZT-ARF when developing authorization models for ZT IoT systems. We introduced the ZT score-based authorization framework (ZT-SAF). We proposed that we need to develop an ongoing authorization policy within the ZT-SAF to meet IoT authorization requirements. Finally, we discussed future directions.

9 Acknowledgment

This work is supported by NSF CREST-PRF Award 2112590 and NSF CREST Grant HRD1736209.

References

[1] Accessed February 2022. Internet of things. https://en.wikipedia.org/wiki/Internet_of_things.
[2] S. Ameer, et al. 2020. The EGRBAC Model for Smart Home IoT. In *(IRI)*. IEEE.
[3] S. Ameer, et al. 2022. An Attribute-Based Approach toward a Secured Smart-Home IoT Access Control and a Comparison with a Role-Based Approach. *Information* (2022).
[4] S. Ameer and R. Sandhu. 2021. The HABAC Model for Smart Home IoT and Comparison to EGRBAC. In *(SAT-CPS)*.
[5] O. Arias, et al. 2015. Privacy and security in internet of things and wearable devices. *TMSCS* (2015).
[6] H. F. Atlam, et al. 2017. Developing an adaptive Risk-based access control model for the Internet of Things. In *(iThings)*. IEEE.
[7] S. Bandara, et al. 2016. Access control framework for api-enabled devices in smart buildings. In *APCC*. IEEE.
[8] N. Baracaldo and J. Joshi. 2013. An adaptive risk management and access control framework to mitigate insider threats. *Computers & Security* (2013).
[9] B. Bezawada, et al. 2018. Securing Home IoT Environments with Attribute-Based Access Control. In *ABAC'18*. ACM.
[10] S. Bhatt, et al. 2017. Access control model for AWS internet of things. In *NISecurity*.
[11] S. Bhatt and R. Sandhu. 2020. Abac-cc: Attribute-based access control and communication control for internet of things. In *SACMAT'20*.
[12] S. Bhatt and R. Sandhu. 2020. Convergent access control to enable secure smart communities. In *(TPS-ISA)*. IEEE.
[13] K. Z. Bijon, et al. 2013. A framework for risk-aware role based access control. In *(CNS)*. IEEE.
[14] Z. B. Celik, et al. 2018. Sensitive Information Tracking in Commodity {IoT}. In *27th USENIX Security Symposium (USENIX Security 18)*.
[15] Z. B. Celik, et al. 2019. IoTGuard: Dynamic Enforcement of Security and Safety Policy in Commodity IoT. In *NDSS*.
[16] M. J. Covington, et al. 2000. *Generalized role-based access control for securing future applications.* Technical Report. Georgia Tech.
[17] T. Denning, et al. 2013. Computer security and the modern home. *Commun. ACM* (2013).
[18] T. Dimitrakos, et al. 2020. Trust aware continuous authorization for zero trust in consumer internet of things. In *TrustCom*. IEEE.
[19] E. Fernandes, et al. 2016. Security analysis of emerging smart home applications. In *SP*. IEEE.
[20] M. Fernández, et al. 2020. A data access model for privacy-preserving cloud-IoT architectures. In *SACMAT'20*.
[21] D. F. Ferraiolo, et al. 2001. Proposed NIST standard for role-based access control. *TISSEC* (2001).
[22] J. Granjal, et al. 2015. Security for the internet of things: a survey of existing protocols and open research issues. *IEEE Comm. Surv. & Tutorials* (2015).
[23] Z. Guoping and G. Wentao. 2011. The research of access control based on UCON in the internet of things. *Journal of Software* (2011).
[24] M. Gupta and R. Sandhu. 2021. Towards Activity-Centric Access Control for Smart Collaborative Ecosystems. In *SACMAT'21*.
[25] G. Ho, et al. 2016. Smart locks: Lessons for securing commodity internet of things devices. In *ASIA CCS '16*. ACM.
[26] V. C. Hu, et al. 2015. Attribute-based access control. *Comp.* (2015).
[27] X. Jin, et al. 2012. A unified attribute-based access control model covering DAC, MAC and RBAC. In *IFIP Annual Conf. on Data and App. Sec.*
[28] S. Kandala, et al. 2011. An attribute based framework for risk-adaptive access control models. In *2011 ARES*. IEEE.
[29] D. R. Kuhn, et al. 2010. Adding attributes to role-based access control. *Computer* (2010).
[30] A. La Marra, et al. 2017. Implementing usage control in internet of things: A smart home use case. In *2017 IEEE Trustcom/BigDataSE/ICESS*. IEEE.
[31] F. Martinelli, et al. 2018. Too long, did not enforce: a qualitative hierarchical risk-aware data usage control model for complex policies in distributed environments. In *CPSS '18*. ACM.
[32] R. McGraw. 2009. Risk-adaptable access control (radac). In *Privilege (Access) Management Workshop. NIST Information Technology Laboratory*.
[33] O. Novo. 2018. Blockchain meets IoT: An architecture for scalable access management in IoT. *IEEE IoT Journal* (2018).
[34] A. Ouaddah, et al. 2017. Towards a novel privacy-preserving access control model based on blockchain technology in IoT. In *Europe and MENA Coop. Adv. in Inf. and Comm. Tech.* Springer.
[35] A. Ouaddah, et al. 2017. Access control in the Internet of Things: Big challenges and new opportunities. *Comp. NW* 112 (2017).
[36] F. Paci, et al. 2018. Survey on access control for community-centered collaborative systems. *ACM Computing Surveys (CSUR)* (2018).
[37] J. Park and R. Sandhu. 2002. Towards usage control models: beyond traditional access control. In *SACMAT '02*. ACM.
[38] J. Park and R. Sandhu. 2004. The UCONABC usage control model. *ACM transactions on information and system security (TISSEC)* (2004).
[39] J. Park, et al. 2011. Acon: Activity-centric access control for social computing. In *ARES*. IEEE.
[40] J. Park, et al. 2021. Activity Control Design Principles: Next Generation Access Control for Smart and Collaborative Systems. *IEEE Access* (2021).
[41] K. K. Patel, et al. 2016. Internet of things-IOT: definition, characteristics, architecture, enabling technologies, application & future challenges. *International journal of engineering science and computing* 6, 5 (2016).
[42] A. Pretschner, et al. 2006. Distributed usage control. *Commun. ACM* (2006).
[43] J. Qiu, et al. 2020. A survey on access control in the age of internet of things. *IEEE Internet of Things Journal* (2020).
[44] A. Rahmati, et al. 2018. Tyche: A risk-based permission model for smart homes. In *2018 IEEE Cybersecurity Development (SecDev)*. IEEE.
[45] S. Ravidas, et al. 2019. Access control in Internet-of-Things: A survey. *Journal of Network and Computer Applications* (2019).
[46] S. W. Rose, et al. 2020. Zero trust architecture. (2020).
[47] R. Sandhu. 1998. Role-based access control. In *Advances in computers*. Vol. 46.
[48] R. Sandhu, et al. 2000. The NIST model for role-based access control: towards a unified standard. In *ACM workshop on Role-based access control*.
[49] R. Sandhu and J. Park. 2003. Usage control: A vision for next generation access control. In *International Workshop on Mathematical Methods, Models, and Architectures for Computer Network Security*. Springer.
[50] R. Sandhu, et al. 2006. Secure information sharing enabled by trusted computing and PEI models. In *ASIACCS '06*.
[51] R. S. Sandhu, et al. 1996. Role-based access control models. *Comp.* (1996).
[52] R. S. Sandhu and P. Samarati. 1994. Access control: principle and practice. *IEEE communications magazine* 32, 9 (1994), 40–48.
[53] B. Tang, et al. 2019. Iot passport: A blockchain-based trust framework for collaborative internet-of-things. In *SACMAT '19*.
[54] Y. Tian, et al. 2017. SmartAuth:User-Centered Authorization for the Internet of Things. In *USENIX Security 17*.