International Journal of Information Security
https://doi.org/10.1007/s10207-022-00639-x

REGULAR CONTRIBUTION

Authentication-enabled attribute-based access control for smart homes

Melike Burakgazi Bilgen[1] · Osman Abul[1] · Kemal Bicakci[2]

Abstract
Smart home technologies constantly bring significant convenience to our daily lives. Unfortunately, increased security risks accompany this convenience. There can be severe consequences when unauthorized or malicious users gain access to smart home devices. Therefore, dependable and comprehensive access control models are needed to address the security concerns. To this end, the attribute-based access control (ABAC) model is usually considered the most satisfactory access control model for running IoT applications. However, the uncertainty left with the authentication stage should be carried to the authorization policy specification. In this work, we extend the ABAC model by carrying the assurance level of user authentication obtained from biometric authentication systems for authorization. The extended ABAC model quantifies how far the authentication matching score is from the predefined threshold. This quantification serves as a regular attribute like others to define authorization policies. The novelty in this quantification is that it consults false matching rate and hence can easily normalize across wide range of biometric authentication devices and algorithms. As a result, the resulting access control policies are concise and easy to comprehend. Moreover, our model is fine-grained in that different access policies can be specified for each smart device functionality. This work also shows, through case studies, that the extended ABAC model is feasible and implementable in XACML language.

Keywords Access control · Attribute-based access control · Internet of Things · False matching rate · Smart home security

1 Introduction

The Internet of Things (IoT) is almost everywhere and its employment is virtually unlimited. IoT applications can be considered in three main categories: smart home, smart health, and smart city [1]. However, in the near future, we will see much more IoT applications in various other areas: including smart buildings, self-driven cars, transportation, industry, agriculture, and energy. An IoT device can be either a sensor or an actuator and can connect directly or indirectly to each other over the Internet. All together they form a huge network consisting of more than 6 billion (as of 2018), more than 14 billion (as of today, 2022[1]) and estimated to be 20 billion (by 2024) devices [2]. Since the Internet of Things consists of many devices communicating with each other, this ecosystem should be well-designed to meet network, system and security needs. IoT consumers need to trust this ecosystem to adopt them in their daily lives as there exists inherent privacy and security risks.

IoT applications are able to collect sensitive information from sensors, users, devices and other systems. Moreover, they can combine public and private data from several cross-devices. Since IoT devices are increasingly becoming part of our daily habits, the lack of strong authentication and access control mechanisms intensify security and privacy concerns. Weak credentials and lack of strong authentication mechanisms are one of the key vulnerabilities of IoT systems [3]. Therefore, properly addressing security and privacy risks in

✉ Melike Burakgazi Bilgen
mbilgen@etu.edu.tr

Osman Abul
osmanabul@etu.edu.tr

Kemal Bicakci
kemalbicakci@itu.edu.tr

[1] Department of Computer Engineering, TOBB University of Economics and Technology, Ankara, Turkey

[2] Informatics Institute, Istanbul Technical University, Istanbul, Turkey

[1] https://iot-analytics.com/number-connected-iot-devices/.

Published online: 16 December 2022

such environments and mitigating them are essential for the sustainable growth.

Among the six security characteristics of confidentiality, integrity, availability, authentication, access control, and non-repudiation [4], this paper focuses on user authentication and access control aspects of IoT security. We aim to build an access control model for home IoT to protect smart home devices from being used by unauthorized or potentially malicious users. Home IoT needs fine-grained, contextual, and dynamic access control to ensure personal data privacy and prevent unauthorized accesses. For this reason, we envision that the attribute-based access control (ABAC) is the most suitable model for home IoT settings. As an observation, smart home users tend to use biometric authentication systems when using smart home devices. In typical biometric authentication systems, the user is either permitted or denied as a whole. However, in our model, a user with a claimed role can be granted/denied for specific functionalities separately based on the authentication assurance level. As a result, access decision policy specification consists of (i) the functionality granularity, (ii) the authentication assurance level, and (iii) the user role.

In our extended ABAC model, after users (with claimed role) send an access request and get authenticated via biometric means, they obtain an authentication matching score (AMS). Then, the FMR (False Match Rate) for the authentication matching scores of the users are computed. The score is named as Access Decision Uncertainty Score (ADUS). The ADUS is added to the user request as a regular attribute. Then, access decision for the requested smart home device functionality is based on the ADUS, the user's role and other environmental attributes. For example, smart home users conveniently use voice assistants like Google Home to manage their smart home devices. In this way, they can easily do online shopping, play music, vacuum clean home by simply saying "Okay Google, vacuum clean home," for instance. Typically, authenticated home users are allowed to access all smart devices in the house with every functionality even if they need to use only specific functionalities. However, in our fine-grained model, for instance, if the ADUS of the authenticated user with the teenager role is no stronger than FMR1000, he cannot perform online shopping but can play music via Google Home Assistant. Whereas if the ADUS is stronger (e.g., at FMR10000) he can perform online shopping or run other critical functionalities. The indicators FMR100, FMR1000 and FMR10000 are the FMR levels at the specificity levels of 1/100, 1/1000 and 1/10000, respectively. Since ADUS is a FMR metric, the smaller the ADUS the stronger the certainty of the user identity verification. The motivation to use ADUS (defined w.r.t. the FMR levels) instead of authentication matching score (AMS) is to normalize across wide range of biometric devices and algorithms for easy comprehension and concise policy specification. The need for functionality-centric access control in IoT is noticeable, but not enough attention by the literature has been paid so far. Limited number of studies mention the importance of a functionality-centric approach in IoT access control [5,6]. Beyond the literature, we go one step further and introduce specifying access decisions based on the functionality's criticality level of *basic*, *important*, and *critical*. Indeed, the criticality level can be treated as a regular attribute like other environmental attributes. This way, the number of access policy rules can be reduced significantly since the access policies are defined per functional criticality level instead of per functionality. To sum up, our study extends the ABAC model by (i) giving access decisions based on the ADUS of authentication matching score, (ii) the role of the user, (iii) and the criticality level of the functionalities. The access decisions can be *permit*, *deny* or *escalate*. We implemented our model in XACML and presented some home IoT scenarios.

This paper is organized as follows. Section 2 explains the importance of access control in smart homes. It discusses smart home access control requirements and explains why ABAC is more suitable for smart home IoT. It presents related work as well. It also provides further discussion for the importance of secure access control for IoT applications. Section 3 presents the proposed AeABAC (Authentication-enabled ABAC) model. First, it describes how to gauge assurance level of user authentication by illustrating it on three scenarios. Next, it gives formal specification of the AeABAC model and presents its feasibility. Then, it presents some smart home scenarios and XACML implementation of our model. Finally, it discusses how our model gives fine-grained access control decisions. Section 4 concludes with some final remarks.

2 The importance of access control for smart homes

Things become smarter by being Internet-enabled and connected to other devices. Although home IoT makes people's lives easier, it uses a significant amount of personal data to work properly [7]. A compromised IoT device/application may gather information from sensors and cross-devices or may use their functionality maliciously [8]. Any unauthorized or malicious access on IoT devices might not affect just a computer or a system, but it might have a severe effect on the physical world [9]. There can be significant consequences in the case when an unauthorized or a malicious user gains access to smart home devices and uses them for their purposes by involving them in the malicious network [2].

Therefore, authentication and access control concepts are the key factors addressing security and privacy issues for in-home IoT [10]. Access control mechanism aims to protect

private data by ensuring only the authorized people access the resources with the privileges defined by access control rules and policies.

2.1 Smart home access control requirements

Access control models in smart homes should take security, privacy, and functionality into consideration [6]. Firstly, they should not define permissions based on the device but each device's functionality. Access permissions for different device functionalities should be created so that a user accesses only required features of the devices regarding "the least privilege" principle of information security. A user should only have the necessary access rights to the device to perform the functionality needed. Due to IoT's connected structure and existing authorization concept, there might be a conflict between what a user thinks an IoT device/application performs and what it does [8]. Two kinds of over privilege have been observed in terms of IoT. One is a coarse capability, and the other one is device-app binding which is implicitly permitting unneeded functionalities [8]. Therefore, home IoT needs fine-grained and precise access control rules. For a secure smart home, access permissions should be defined based on the device's capabilities rather than based on the device as a whole.

Multi-user accesses for multi-device scenarios need to be considered as well. In smart home platforms, it is common to give access to all devices in the smart home environment. However, smart home users feel uncomfortable at defining the same access permissions to other users like Spouse, Child and Guest. The survey performed in the study [6] shows the access decision requirements change according to different capabilities of the devices for different user roles. For example, parents do not want their small children to perform online shopping via voice assistant; on the other hand, they do not see any problem with playing music or turning on/off lights [6]. In one study, parents were asked whether they preferred monitoring their teens using smart locks and cameras. Most parents chose to monitor their teens either through unrestricted access to logs or through real-time access notifications. As a result, multi-user access control rules for different functionalities of the devices in the home IoT should be supported. Access permissions for each function should be specified according to its risk category and the consequences of unauthorized or malicious accesses [6].

Because smart homes consist of several types of devices and have dynamic nature in terms of users and environments, they need context-aware, flexible, and fine-grained access control policies [11]. Detailed access rules for multiple users for different functionalities of devices should be considered to make access control policies stronger.

2.2 Why ABAC?

First of all, access control mechanisms ensure the authorization on the resources [12]. The traditional access control methods are Discretionary Access Control (DAC), Mandatory Access Controls (MAC), and Role-Based Access Control (RBAC). MAC model provides multi-level security. Each resource has a confidentiality level. Users can access resources if it is permitted for the respective confidentiality level [13]. In DAC, a resource owner defines who can access the resource with which rights. For example, in operating systems, the file owner determines access permissions for his file. This model offers an elementary level of security [13]. RBAC model grants permissions based on the roles of the users. It uses role-permission, user-role, and role-role relationships to make access decisions [7]. Even though it is helpful in small environments, it may not meet IoT access control needs due to its user-centric and static structure. Overall, the management efficiency of traditional access control models is lower in terms of IoT requirements. It is hard to implement detailed access control scenarios in these models and apply the least privilege principle [7]. However, IoT needs a dynamic, fine-grained, and beyond user-centric access control model.

Unlike the traditional access control models, ABAC does not define access permission between subjects and resources. Access permissions are defined based on the attributes. This makes the ABAC model more flexible and dynamic, thus more suitable for home IoT. New permissions can easily be granted to attributes as new resources, contextual information and actions are arrived to the existing system. ABAC can convert the policies and rules into permissions dynamically by retrieving attributes of the request. It needs to consider all of the attributes due to subjects, resources, and environmental conditions like current time and location. It should respond to the changes in the attribute values of the request. Attributes allow us to create access policies corresponding to real-life scenarios to adapt to environmental changes. Subject attributes further define the subject that requests a resource. It can be age, role, department, management level, etc. They can be static (like roles) or dynamic (like age, time and weather). A resource is managed by ABAC rules and policies and its attributes, for instance, are resource type, sensitivity, resource feature and classification. Finally, environmental attributes can be about location, time, weather, and other dynamic attributes like threat level or risk level [14].

The high-level ABAC definition depicted in the NIST ABAC guide [14] is redrawn in Fig. 1. From the figure, the ABAC model consists of Policy Enforcement Point (PEP), Policy Decision Point (PDP), Policy Information Point (PIP), and Policy Administration Point (PAP). The subject's request is met by PEP. PEP sends it to PDP to make an authorization decision. PDP evaluates the subject's request by finding

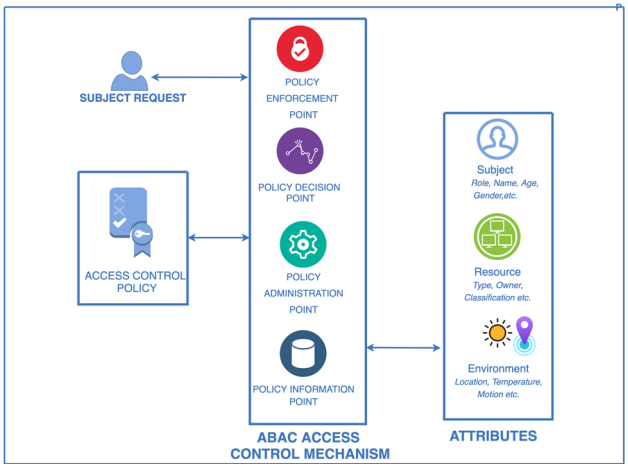


Fig. 1 High-level definition of ABAC model

corresponding access policies. If additional attributes are needed, PDP requests more attributes about the request from PIP. PDP then makes an authorization decision and forwards it to PEP. According to the decision returned from PDP and obligations, PEP permits or denies access to the resource [15].

The number of different access condition combinations in ABAC is much more than the traditional access control models [13]. Due to the variety of the attributes, it is possible to define fine-grained access control rules and policies in contrast to traditional access control models [10]. Access policies and rules consist of different subject, objects (as resources), and environmental attributes. Also, with the help of environmental conditions such as location, date-time, threat levels and IP address, the ABAC model can support dynamic and contextual access control rules [13].

Supporting interoperability is the another reason making ABAC more convenient for home IoT. As long as attributes of unknown users meet the criteria of some existing policies and rules, the user gains access to the related resource [10].

To summarize, because of its dynamic nature, home IoT needs an access control mechanism that is dynamic, context-aware, beyond user-centric, and supports fine-grained policies. Therefore, the ABAC model is a good match by creating flexible and dynamic access control rules through attributes. Access permissions are defined for attributes, not for identities. It is not user-centric; instead, it establishes allowable operations for a set of attributes. Environment attributes make it context-aware by getting information about the current time, location, sensors, threat level, IP address and others.

2.3 Related work

There have been several models proposed to improve access control for IoT. Some of them offer extensions to ABAC [16–18] and some of them suggest using ABAC for IoT [19,20]. Other studies focus on secure accessing of sensitive data in IoT, thus proposing blending blockchain technology and ABAC model [21,22]. Song et al. [22] offer to use smart contracts for access control decision-making to implement a trustworthy access control model. Aghili et al. [21] proposes Multi-Level Security ABAC (MLS-ABAC) scheme that uses an authorized encryption strategy to safeguard the data integrity, in addition to taking security-level verification and dynamic features into account. While [21] enhances the security and privacy of IoT systems by adding a security-level verification before partial decryption, our model improves the security level of authentication.

Cathey et al. [23] uses edge-centric access control structure in IoT by suggesting a Tag-Based Access Control design that focuses on splitting data into many digital twins. Giving access to specific subsets of data within shadows is a sort of tag-based access control since the data in a given shadow are directly tied to the tags applied to that data. Goyal et al. [24] present a prototype for using ABAC for in-home IoT applications. They define policies at the device level by evaluating users, devices, and operation attributes. Our model is different in terms of defining policies at a device's functionality level. In addition, [25] presents an Activity-Centric Access Control (ACAC) model for smart ecosystem. An activity is defined as functions, i.e., capabilities that different devices can perform at a certain time. A device can perform one or multiple activities. Activities can be related to each other or need to be performed in a particular order. They evaluate environmental conditions, obligations, and mutability of activities of the attributes in access decisions. The proposed model can be considered as a fine-grained functionality-based access control model. Later in [26], Sandhu et al. take a step forward and defines components of the ACAC presented in [25] and compare it with other models. Even though they care about the need for a fine-grained access control model for IoT, we have not come across any work using the assurance level of user authentication from biometric authentication systems during access decisions. In the study [16], the certainty of the user is considered, but it is calculated via behavior anomaly techniques. Similarly, in the study [17], some machine learning techniques are used to get the probability of risk associated with the user. Since it's approach of machine learning depends on large login logs, it is not cold-start free. Our work is the first that considers the assurance level of user authentication from statistics (through bounded FMR) point of view and granting fine-grained permissions for device functionalities accordingly.

The need for functionality-centric access control in IoT is noticeable, but not enough attention has been paid so far. Some studies mention the importance of a functionality-centric approach in IoT access control [5,6]. He et al. [6] emphasizes the gap that the authentication in a smart home

Table 1 Locating our proposal of AeABAC within the literature

Ref nos.	Access control model	Assurance level	Cold start	Functionality centric	Functionality categorization	Bounded FMR
[16]	ABAC	Yes	No	No	No	No
[17]	Adaptive Risk Aware ABAC	Yes	No	No	No	No
[25]	Activity Centric Access Control	No	NA	Yes	Partial	No
[5]	Functionality Centric Access Control	No	NA	Yes	No	No
[6]	NA	No	NA	Yes	Partial	No
[26]	Activity Centric Access Control	No	NA	Yes	Partial	No
[27]	ABAC	No	NA	No	No	No
[23]	Tag Based Access Control	No	NA	No	No	No
[24]	ABAC	No	NA	No	No	No
[22]	ABAC	No	NA	No	No	No
[21]	Multi-Level Security ABAC (MLS-ABAC)	No	NA	No	No	No
Our model	AeABAC	Yes	Yes	Yes	Yes	Yes

is based on a single user per device. They performed a user study and found that smart home users prefer different access control policies for different functionalities of a single device. They also take into consideration who uses which functionality of the device under which conditions such as the location of the user, weather, and time. Unfortunately, [6] does not relate this approach to any access control model. Zeng and Roesner [27] designed smart home application by addressing multi-user smart homes security and privacy needs mentioned in [6] and other studies. While their approach includes role-based, location-based, supervisory, and reactive access controls, they do not take into account the uncertainty of the user's real identity during authentication.

Table 1, in summary, locates our proposal of AeABAC within the literature and contrasts it with other studies along five dimensions: Assurance Level of User Authentication, Cold Start (i.e., whether the assessment of uncertainty of user identity requires large login logs), Functionality-centric, Functionality Categorization, and Bounded FMR.

3 AeABAC model

This paper is an extended version of the paper "Extending Attribute-Based Access Control Model with Authentication Information for Internet of Things" published in the 2020 International Conference on Information Security and Cryptology (ISCTURKEY). In the previous work [28] our approach was giving access permissions based on several threshold values. Even if we mentioned the difficulty of choosing several thresholds and suggested using performance metrics instead, it was left as a future work. In this paper, we focus on using performance metrics instead of defining several threshold values. Our contribution is discussing performance metrics deeply and choosing FMR of authentication matching score. As a result, access control

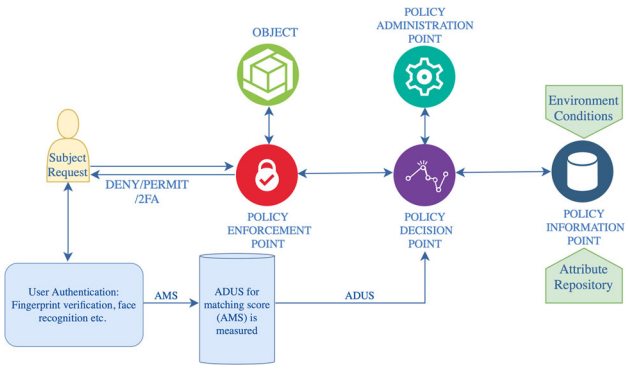


Fig. 2 Our AeABAC model as an extension of ABAC model

policies are defined in terms of the ADUS and the functionality grouping. We group device functionalities based on their criticality level as basic, important, and critical and give access decisions accordingly. As a result, we end up with a more concise and convenient model especially for IoT applications where the user authentication is through biometric means. The AeABAC model, as an extension of ABAC, is shown in Fig. 2.

3.1 Assessing the assurance level of user authentication

In smart home environment, physical and behavioral biometric authentication methods are expected to be used instead of traditional authentication methods. Physical biometrics is about measurements of the human body signatures. Face recognition, hand geometry, fingerprint recognition, and DNA matching are some examples of physical biometrics. On the other hand, behavioral biometrics are associated with users' unique movements or habits. Keystroke dynamics, voice recognition, and signature recognition are some of the behavioral biometrics methods. Among them voice

recognition and fingerprint recognition are the most popular biometric means used in IoT applications [29].

Biometric systems use algorithms to decide whether to permit or deny access requests. They calculate authentication matching score (AMS) of the user by executing preprocessing, scoring, and thresholding phases to make the authentication decision [30]. The AMS is obtained by measuring the similarity between collected samples and the presented sample. If the score is above a predefined threshold, a match decision is made; otherwise, a non-match decision is made [31]. There are two types of error rates: *False Match Rate* (False Accept Rate) and *False Nonmatch Rate* (False Reject Rate). If the imposter's sample is accepted as a match for a legitimate user's template, it is called a False Match. On the other hand, if the legitimate user's sample is declared as a nonmatch, it is called a False Reject. The error rates depend on the threshold value, i.e., they change as the threshold value changes [32]. As a result, deciding on the threshold value is crucial for granting accesses to the legitimate users. As stated in [30], the threshold value selection is a tradeoff between the two error types.

As discussed, the access request is granted only if the AMS is above the threshold. In our study, rather than using the AMS directly for access decision, we propose to transform this score so that it is comparable with FMR to evaluate the assurance level and give access permissions accordingly. The next question is how to obtain FMR levels for the specificity of devices and algorithms. This information can be obtained through (i) collecting large login samples during operation, (ii) manufacturer specifications given in the white papers, or (iii) benchmarking services. As an example in this paper, we will employ the reported results for the benchmark FV-HARD-1.0 of FVC-onGoing project [33,34] which offers benchmarks with specific datasets and testing protocols for several biometric algorithms such as fingerprint verification [35]. It compares the performance of several algorithms on the same benchmark and publishes the statistical performance results. For the fingerprint verification task, for instance, the False Non Match Rate (FNMR) and False Match Rate (FMR) are computed at different threshold values. To compute the FNMR and FMR, genuine (matching two fingerprints from the same finger) and impostor (matching two fingerprints from different fingers) attempts are made [35]. The genuine and the impostor matching score distributions are computed and plotted as a histogram at the FVC-onGoing project website [33]. In fingerprint verification, higher matching scores are associated with more closely matching images. They obtain score distributions for impostors and genuine at the thresholds between 0.0 and 1.0. Among other score distributions, FMR and FNMR scores are computed for the threshold t, ranging from 0.0 to 1.0. Then Decision Error Tradeoff (DET) graph is plotted as well, where the pairs (FMR(t), FNMR(t)) are plotted along the two axis [35]. The DET graph shows the lowest FNMR at the specified FMR levels.

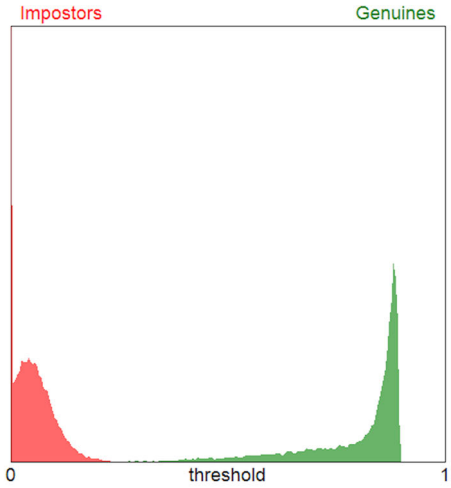
For a comprehensive and concise policy specification that does not change from device to device and algorithm to algorithm we need a normalization. At this point, FMR is a good choice as it gives how far the AMS is from the threshold value. Indeed, there may be significant FMR differences for seemingly similar AMSs. Likewise, seemingly dissimilar AMSs on different biometric systems may yield similar distances from the respective thresholds, but again exhibit significant dissimilar FMRs. To be more concrete, we show them through three scenarios.

3.1.1 Scenario I

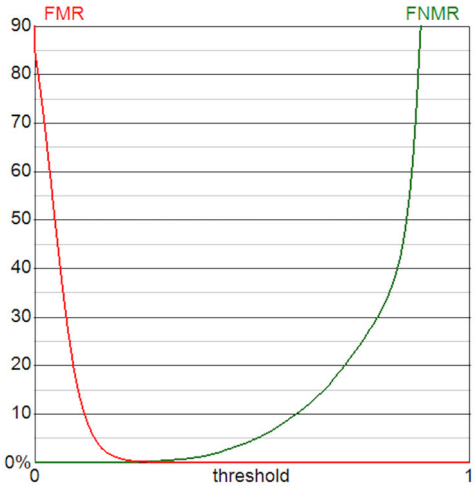
In the first scenario, we assume there are two users, Bob and Mary. They use the same smart device which authenticates users via a fingerprint reader that uses the fingerprint matching algorithm HXKJ V4.3 on FV- HARD-1.0 presented in the FVC-onGoing project [33]. Bob tries to unlock the smart door using his finger and gets an AMS of 0.8. Later, Mary tries to open the smart door via fingerprint and gets an AMS of 0.7. Suppose the threshold T is picked as to satisfy FMR(T) = FMR110. We want to analyze what differs if a user gets AMS of 0.7 or 0.8. To clarify this point, we use the performance results (shown in Fig. 3) of fingerprint matching algorithm HXKJ V4.3 on FV- HARD-1.0 [33,34]. We propose to make access decisions based on the FMR of the matching scores. The FMR and FNMR for the AMS of 0.7 and the AMS of 0.8 should be calculated to understand the certainty level of users. The DET graph plotted for this algorithm provides the necessary information. It shows the corresponding FNMR values when FMR is 0.01, 0.001, and 0.0001. We assume when the AMS is $\approx$ 0.7, the FMR is close to FMR(T+0.1). When the AMS is $\approx$ 0.8, the FMR is close to FMR(T+0.2). Even though there is a slight difference between the two AMSs, the corresponding FMR values for the two scores differ significantly (one order of magnitude) as shown in Fig. 4.

3.1.2 Scenario II

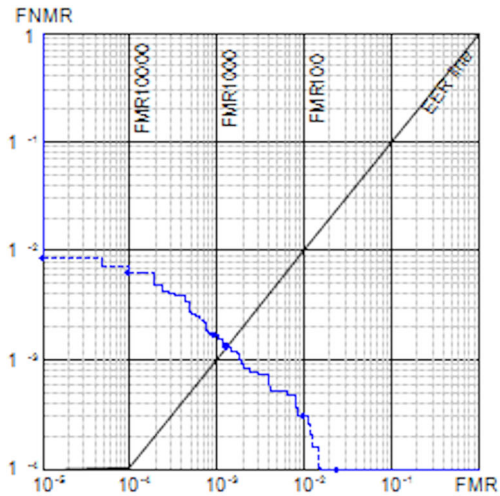
We assume the single-user Bob uses two smart devices using different biometric authentication algorithms in this scenario. The first one is the smart door lock as in the first scenario that uses fingerprint algorithm HXKJ V4.3 on FV-HARD-1.0, and the other one is a smart camera using the fingerprint matching algorithm MMFV 12.0 on FV-HARD-1.0 [34]. The score histogram, match rate graph, and DET graph of the MMFV 12.0 algorithm are given in Fig. 5 [33,34]. Two systems can have different confidence levels for the same user. For example, Bob gets an AMS of 0.7 in the smart lock so the corresponding FMR is FMR(T+0.2) where FMR(T) is equal to FMR110. On the other hand, he receives an AMS of 0.4



(a) Histogram for HXKJ V4.3



(b) Match Rate Graph for HXKJ V4.3



(c) DET Graph for HXKJ V4.3

Fig. 3 HXKJ V4.3 metrics [33]

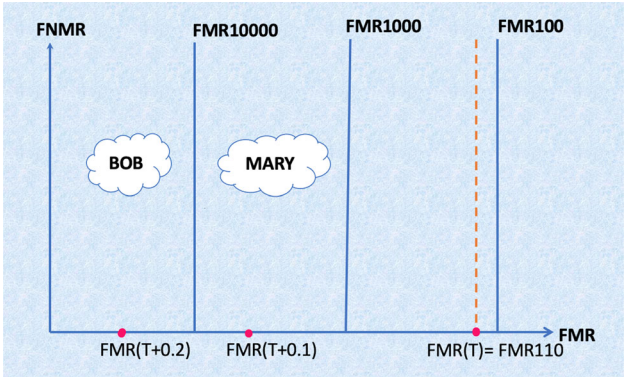


Fig. 4 FMRs for the Scenario I

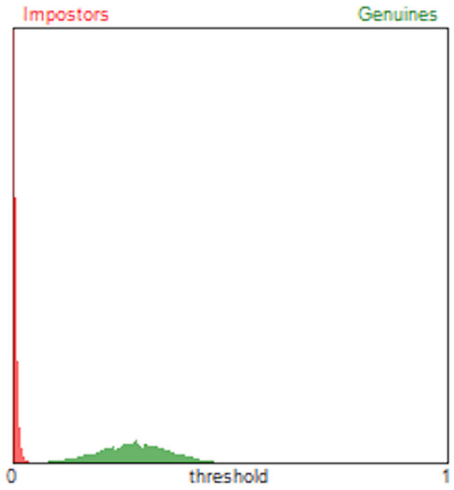
when authenticating at the smart camera. The corresponding FMR is around FMR(T+0.2) where threshold FMR(T) is equal to FMR120, which is good for matching algorithm MMFV 12.0. When comparing the two AMSs, the uncertainty left with the Bob's identification is not the same, even though the matching scores are 0.2 points ahead of the respective threshold T (Fig. 6). Therefore, access decisions based on FMR provides us with a more accurate certainty level of authentication.

3.1.3 Scenario III

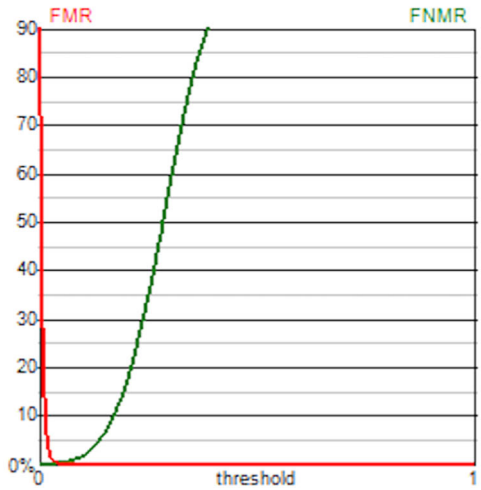
This scenario is similar to the Scenario I. Again there are two users, say Eric and Jane, but they use palmprint verification for the authentication. This time, they both utilize the same smart security camera, which authenticates users via a palmprint verification that uses the algorithm HXKJ 3.02 on PV-FULL-1.0 presented in the FVC-onGoing project [34]. Suppose they are successful at their attempt to access the smart camera with AMSs of 0.5 (Eric) and 0.6 (Jane). Suppose the threshold T is chosen so as to satisfy FMR(T) = FMR120. We are interested in how far the AMSs of 0.5 or 0.6. To demonstrate this point, we use the performance results of the palmprint verification algorithm on PV-FULL-1.0 [33,34]. The score histogram, match rate graph, and DET graph for the HXKJ 3.02 algorithm are given in Fig. 7 [33,34]. Those plots are informative to answer our point of interest. Aligning the two AMSs, with the AMS of 0.5 the FMR is close to FMR(T+0.1) and with the AMS of 0.6 the FMR is close to FMR(T+0.3). Even if the two AMSs slightly differ, we see from Fig. 8 that there exists an order of magnitude difference in terms of the FMR.

3.2 AeABAC for smart homes

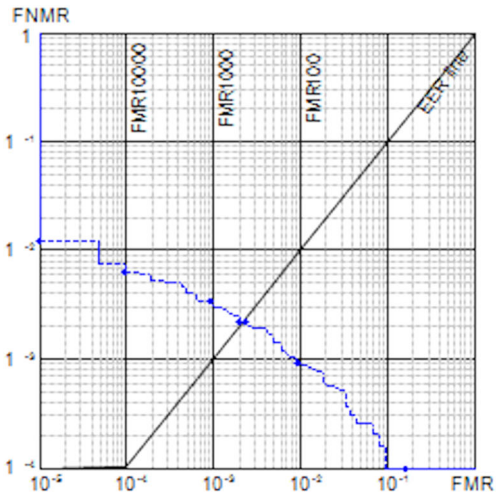
This section defines our AeABAC (Authentication-enabled Attribute-based Access Control) model, which is intended for user-to-device interaction in smart homes. We propose a for-



(a) Histogram for MMFV 12.0

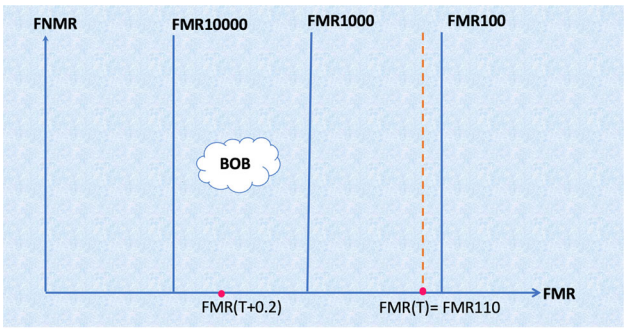


(b) Match Rate Graph for MMFV 12.0

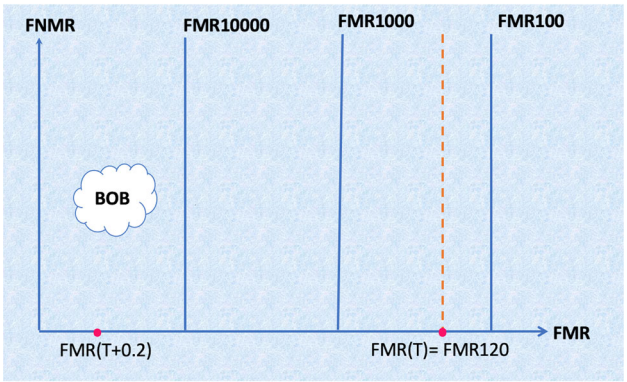


(c) DET Graph for MMFV 12.0

Fig. 5 MMFV 12.0 metrics [33]



(a) System 1



(b) System 2

Fig. 6 FMRs for the Scenario II

mal specification of the AeABAC model based on the basic framework presented in the study [36]. Our proposal extends Ameer et al. [36] by introducing the ADUS as an important regular attribute. The basic components of AeABAC model include: Users (U), Sessions(S), Devices(D), Environment Situations (ES), and Policies(P).

3.2.1 Definitions

Users(U): Set of entities that perform operations on devices.

Sessions(S): Set of all user sessions that are created and ended by users during they perform action on devices. Each session is associated with a user who manages the session.

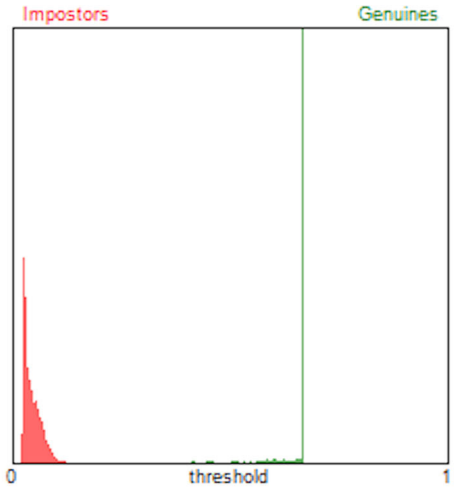
Devices(D): Set of smart home devices like smart door locks, camera, and voice assistants.

Device Functionalities(DF): Set of functionalities that users may perform on devices as defined by the device manufacturer.

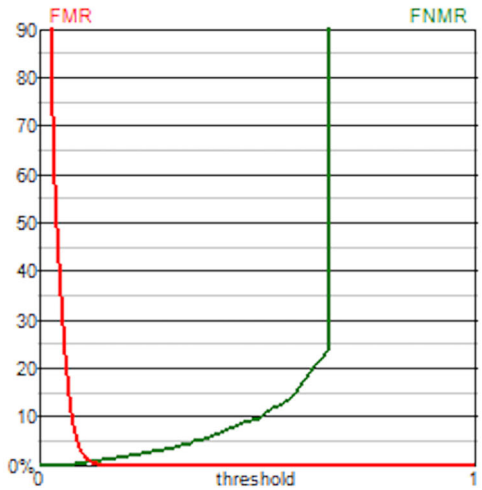
Functionality Types: An atomic attribute that describe the criticality level of device functionalities such as basic, important, critical.

Authentication Matching Score(AMS(u)): A user attribute with a value of 0.0 to 1.0 that is calculated by the biometric system that authenticates the user.

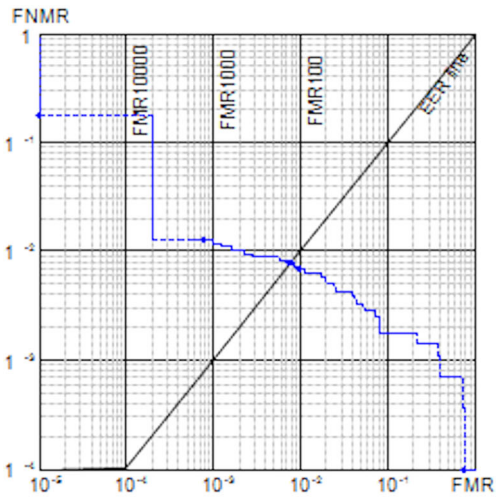
Access Decision Uncertainty Score(ADUS(u)): A function that takes the user authentication matching score as input



(a) Histogram for HXKJ 3.02



(b) Match Rate Graph for HXKJ 3.02



(c) DET Graph for HXKJ 3.02

Fig. 7 HXKJ 3.02 metrics [33]

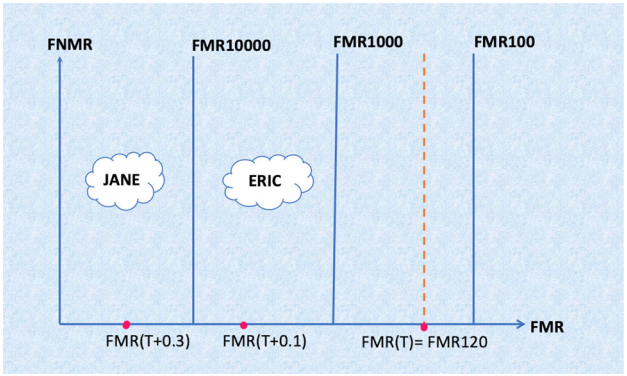


Fig. 8 FMRs for the Scenario III

and gives FMR for the AMS for the user. It returns the result of FMR(AMS(u)).

Environment Situations (ES): Set of environment situations that represent the current state at a specific point in time such as IP address, temperature, time, or weather.

3.2.2 Attributes

Users, sessions, devices, environment situations, device features, and operations have properties that are called attributes. The set all of attributes consist of User Attributes (UA), Session Attributes (SA), Device Attributes (DA), Device Feature Attributes (DFA), Operation Attributes (OA), and Environment Situation Attributes (EA). Attributes are functions that take an entity and returns a specific value from its range. They may have either atomic or set value. A finite collection of atomic values defines an attribute range. A set valued attribute returns a subset of the range, but an atomic valued attribute returns one value from the range.

Each attribute $att \in UA \cup SA \cup DA \cup DFA \cup ESA$ has a finite set of atomic values, denoted with $Range(att)$. The property $attType(att) \in \{set, atomic\}$ specifies attributes to be set or atomic valued.

User Attributes (UA): The set of user attributes for users. Role, age, and gender are example of user attributes. Each attribute function att_u in UA, maps users in U to attribute values.

att_u :U $\rightarrow 2^{Range(att_u)}$ if $attType(att_u)$= set

att_u :U $\rightarrow Range(att_u)$ if $attType(att_u)$= atomic

Session Attributes (SA): The set of subject attributes for subjects. They inherit the attributes of the user who created the session, formally $SA(s) \subseteq UA(sessionUser(s))$ for each $s \in S$. Each attribute function $att_s \in SA$ maps sessions in S to attribute values.

att_s :S $\rightarrow 2^{Range(att_s)}$ if $attType(att_s)$= set

att_s :S $\rightarrow Range(att_s)$ if $attType(att_s)$= atomic

Device Attributes (DA): The set of attributes that define smart devices such as living room devices. They are partial

functions so some devices may have no attributes assigned. Each attribute function $att_d \in DA$ maps each device in D to attribute values.

att_d :D $\rightarrow 2^{Range(att_d)}$ if $attType(att_d)$= set

att_d :D $\rightarrow Range(att_d)$ if $attType(att_d)$= atomic

Device Functionality Attributes (DFA): The set of attributes that define smart device functionalities such as turning on/off, online shopping, playing music and lights on/off. Each attribute function $att_{df} \in DFA$ maps each device functionalities in DF to attribute values.

att_{df} :DF $\rightarrow 2^{Range(att_{df})}$ if $attType(att_{df})$= set

att_{df} :DF $\rightarrow Range(att_{df})$ if $attType(att_{df})$= atomic Also, each device functionality has a type such as basic, important and critical.

Environment Situation Attributes (ESA): Set of attributes that represent current environment situation such as temperature, time or weather. Each attribute function $att_{es} \in ESA$ maps environment situations in ES to attribute values.

att_{es} :ES $\rightarrow 2^{Range(att_{es})}$ if $attType(att_{es})$= set

att_{es} :ES $\rightarrow Range(att_{es})$ if $attType(att_{es})$= atomic

Policies(P): Authorizations on the basic components can be specified through policies or authorization rules. Policies are boolean functions that take session, device, and environment attributes for the instance and evaluate if the requested operations should be permitted.

Access Decision: The access decision takes six inputs that are session user $u \in U$ with the authorization decision score ADUS(u) during the session $s \in S$ is permitted to perform an operation for the device functionality $df \in DF$ on the device $d \in D$ at the environment situations $es \in ES$ if ADUS(u) satisfy isAllowable(s:S, ADUS(u), es:ES, d:D, df:DF). Therefore each access decision is associated with an attribute-based authorization policy that determines whether a user should get that permission on a device. An authorization policy compares the necessary session, environment, device, device feature attributes, and ADUS for the user to make the access decision.

3.2.3 Examples

We describe some example rules to show the components and configurations of AeABAC model. In the first rule, the roles of parent and teenager with ADUS stronger than FMR1000 can access important functionalities of any device within the home. In the second rule, all users inside the home with ADUS at least FMR100 can perform basic device functionalities. We can define and configure the AeABAC model and specify the rules as follows:

U = { Tracy, Bob, Meggy}

UA = { Role, Location}

Role(u) : U $\rightarrow$ {parent, child, teenager, babysitter, guest }

Location(u) : U $\rightarrow$ {inside, outside}

Role (Tracy) = parent

Role (Bob) = teenager

Role (Meggy) = babysitter

SA = {Role, Location}

D= {GoogleHomeAssistant, PhilipsHueLamp, AndroidBox, DoorLock, Camera}

$DF_{GoogleHomeAssistant}$ = {OnlineShopping, PlayingMusic, TurningOn, TurningOff }

$DF_{AndroidBox}$ = {NetFlix, Youtube, Spotify, PlayGame}

$DF_{PhilipsHueLamp}$ = {ON, OFF}

$DF_{DoorLock}$ = {Open, Close}

DF_{Camera} = {Open, Close,ChangeAngle, ViewRecords}

DFA={Type}

Type(df): DF $\rightarrow$ { Basic, Important, Critical}

Type(OnlineShopping) = Type(ChangeAngle) =

Type(ViewRecords) = Critical

Type(ON) = Type(OFF) = Type(PlayGame) =

Type(Open) = Type(Close) = Important

Type(PlayingMusic) = Type(NetFlix) = Type(Youtube) = Type(Spotify) = Basic

ES= { Now}

ESA= { day}

day(es): ES $\rightarrow$ {S,M,T,W,Th,F,Sa}

We can express the scenarios mentioned above formally as:

The first rule: isAllowable(s:S, ADUS(u), es:ES, d:D, df:DF) = (Role(s) = teenager $\vee$ Role(s) = parent $\wedge$ $ADUS(u) < FMR1000 \wedge$ (Type(df) = Important))$\wedge$ Location(s)=inside)

The second rule: isAllowable(s:S, ADUS(u), es:ES, d:D, df:DF) = ($ADUS(u) < FMR100 \wedge$ Type(df) = Basic $\wedge$ Location(s)=inside)

3.3 Functionality-based access decisions

Defining functionality-based access control policies is essential to ensure the most secure access control. It is also vital to guarantee that the person granted with access is the one who prevent the device from being used maliciously [6]. In our model, measuring the ADUS for an authentication score is essential to ensure the subject's identity. As a result, we categorize access decisions into three types of functionalities as basic, important, and critical. Basic functionalities are not considered risky in case of unauthorized access. They may be vacuuming, turning lights on, turning on the TV, or playing music. Important functionalities are considered risky in case of malicious or unauthorized accesses happen. Examples include locking the door, turning on the oven, or turning the lawnmower on. Lastly, critical functionalities are considered most risky in case of unauthorized accesses are granted. Examples include turning on/off camera, online shopping, and deleting access logs. Note that even the important and critical functionalities should be treated differently. Besides

CRITICAL FUNCTIONALITIES	ADUS<=FMR10000	FMR10000<ADUS<= FMR1000	ADUS>FMR1000
Spouse	Permit	Escalate (2FA)	Deny
Teenager	Permit	Escalate (2FA)	Deny
Child	Deny	Deny	Deny
Babysitter	Deny	Deny	Deny
Guest	Deny	Deny	Deny

IMPORTANT FUNCTIONALITIES	ADUS<=FMR10000	FMR10000<ADUS<= FMR1000	ADUS>FMR1000
Spouse	Permit	Escalate (2FA)	Deny
Teenager	Permit	Escalate (2FA)	Deny
Child	Deny	Deny	Deny
Babysitter	Permit	Escalate(2FA)	Deny
Guest	Permit	Escalate(2FA)	Deny

BASIC FUNCTIONALITIES	ADUS<= FMR100
Spouse	Permit
Teenager	Permit
Child	Deny
Babysitter	Permit
Guest	Permit

Fig. 9 Tabular representation of access decisions rules

CRITICALITY LEVEL OF FUNCTIONALITY	NAME
Critical	☼ Online Shopping ☼ Turning Camera On/Off
Important	☼ Switch On/Off ☼ Administrative Features
Basic	☼ Playing Music ☼ Lights On/Off

Fig. 10 Tabular representation of criticality level for functionalities

permit and deny, we also allow the *escalate* to be specified for manual privilege escalation.

A nice feature of our AeABAC model is that simple rules can be edited conveniently using a spreadsheet as shown in Fig. 9. In the figure, there are separate tables for each criticality level, and within each table the rows correspond to roles and columns correspond to the various uncertainty levels. Likewise, as shown in Fig. 10, the functionalities can be assigned with criticality levels. The tabular rule editing brings a significant convenience for policy specification.

3.3.1 Feasibility

We aim to make the smart devices' authorization accurately and efficiently. For assessing our model's feasibility, we assume there are n different biometric devices in our smart home environment. Since the authorization decisions are based on criticality levels and ADUS, within the AeABAC model it suffices to define only one rule for each criticality level as shown in Fig. 9. Recall that the metric ADUS normalizes the uncertainty levels across all of the devices and there is no need for a separate rule definition for each biometric authentication device. Since the same ADUS score for each device are likely to be obtained for distinct values of AMSs, we need to repeat the tables in Fig. 9 for each device if the same effect is desired in terms of AMSs.

CRITICAL FUNCTIONALITIES DEVİCE 1	AMS>=0.85	0.85>AMS>=0.7	AMS<0.7
Spouse	Permit	Escalate (2FA)	Deny
Teenager	Permit	Escalate (2FA)	Deny
Child	Deny	Deny	Deny
Babysitter	Deny	Deny	Deny
Guest	Deny	Deny	Deny
CRITICAL FUNCTIONALITIES DEVİCE 2	**AMS>=0.6**	**0.6>AMS>=0.5**	**AMS<0.5**
Spouse	Permit	Escalate (2FA)	Deny
Teenager	Permit	Escalate (2FA)	Deny
Child	Deny	Deny	Deny
Babysitter	Deny	Deny	Deny
Guest	Deny	Deny	Deny
CRITICAL FUNCTIONALITIES DEVİCE 3	**AMS>=0.4**	**0.4>AMS>=0.2**	**AMS<0.2**
Spouse	Permit	Escalate (2FA)	Deny
Teenager	Permit	Escalate (2FA)	Deny
Child	Deny	Deny	Deny
Babysitter	Deny	Deny	Deny
Guest	Deny	Deny	Deny
CRITICAL FUNCTIONALITIES DEVİCE 4	**AMS>=0.7**	**0.7>AMS>=0.55**	**AMS<0.55**
Spouse	Permit	Escalate (2FA)	Deny
Teenager	Permit	Escalate (2FA)	Deny
Child	Deny	Deny	Deny
Babysitter	Deny	Deny	Deny
Guest	Deny	Deny	Deny
CRITICAL FUNCTIONALITIES DEVİCE 5	**AMS>=0.5**	**0.5>AMS>=0.4**	**AMS<0.4**
Spouse	Permit	Escalate (2FA)	Deny
Teenager	Permit	Escalate (2FA)	Deny
Child	Deny	Deny	Deny
Babysitter	Deny	Deny	Deny
Guest	Deny	Deny	Deny

Fig. 11 Tabular representation of access decisions rules using AMSs for five distinct biometric devices

Table 2 The number of rules within the access policies

# of devices	ABAC (using AMS)	AeABAC (using ADUS)
5	$n \times 35 = 175$	35
n	$n \times m$	m

For a particular configuration, suppose we have $n = 5$ different biometric devices. The same access decisions for critical functionalities shown in Fig. 9 (which defines rules in terms of ADUS) need to be repeated for five distinct devices as shown in Fig. 11 (which defines rules in terms of AMS). The reason for the repeat for each device is that the same ADUS is obtained at different AMSs for the five devices. For instance, from the figure, the ADUS = FMR10000 is obtained at AMS = 0.85 for Device 1 and at AMS = 0.6 for Device 2, and so on. As a result, AeABAC attains comprehensive, concise and efficient access policies.

Table 2 shows how concise the AeABAC model in comparison to basic ABAC models, for the particular example given in Fig. 9 (35 rules) and the scenario expressed in Fig. 11 (5 devices, 175 rules), and for the general case of m rules and n devices. Clearly, the number of rules is n times smaller with AeABAC.

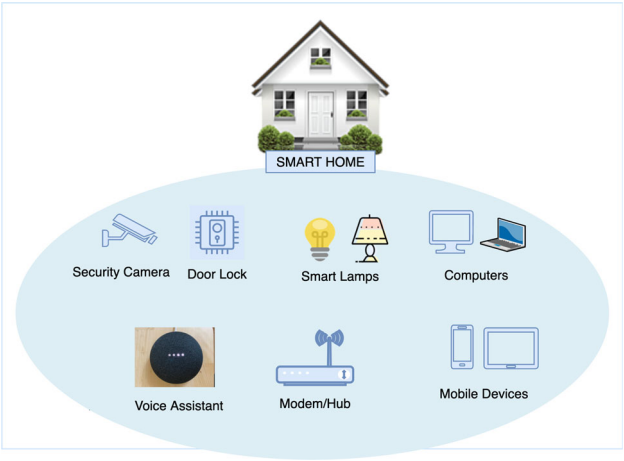


Fig. 12 Sample smart home environment

3.4 Sample smart home scenarios and XACML implementation

In this scenario, we have Google Home Assistant, Philips Hue Smart Lock, Xiaomi Mi Home Security Camera in our small home environment, as visualized in Fig. 12.

As the policy ingredients, we have following components. The user profiles are spouses, children, teenagers, babysitters, and guests. We use Security Policy Tool to implement our scenarios in Extensible Access Control Markup Language (XACML), an attribute-based access control standard. Subject, Resource, Environment and Actions are defined as shown in Figs. 13 and 14. We define the ADUS levels as *strong* (where $ADUS \leq FMR10000$), *good* (where $FMR10000 < ADUS \leq FMR1000$) and *weak* (where $FMR1000 < ADUS \leq FMR100$), and *low* (where $FMR100 < ADUS$). Policies for permit and deny rules for each criticality levels can be expressed under different subsections for the sake of modularity (Fig. 15). Using the components, a sample XACML policy specification with four permit rules projected on critical functionalities is given in the Appendix.

4 Conclusion

Smart home environments consist of various devices and users, so it needs a dynamic, fine-grained, and context-aware access control models. This paper discussed access control models in the Internet of Things and proposed our extended version of the ABAC model, named as AeABAC. We aim to extend the ABAC model to a fine-grained access control model by offering to use access decision uncertainty scores in addition to subject, object, and environmental attributes in access control rules and policies to make more detailed and granular access decisions. To decide how to measure authen-

Fig. 13 Subject, resource, and actions attributes

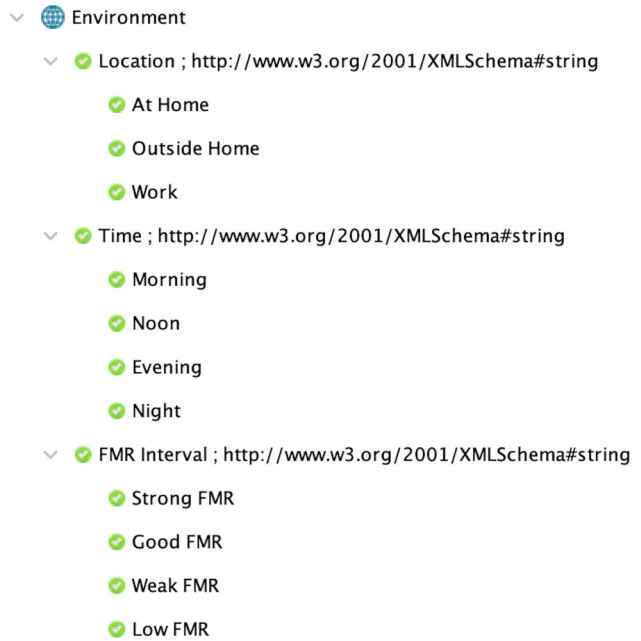


Fig. 14 Environment attributes

Fig. 15 Smart home ABAC policies

tication score, we investigated some performance metrics. Performance metrics are functions of thresholds, while in our model, they are functions of matching scores, in particular function of false matching rate (FMR). We use FMR performance metrics as a regular attribute during access policy specifications. This way we achieve normalized and easy to comprehend metrics that quantifies the uncertainty on user identification from the biometric authentication devices. The feasibility study of AeABAC has shown that it attains concise and more comprehensive access policies. We have also shown that the AeABAC model can be implementable in XACML through a smart home IoT case study. Applicability of AeABAC beyond home IoT remained as a future work. In another future work, we plan to investigate how privacy issues relate to the access decisions.

Data availability No datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors declare no competing financial interests.

Ethical approval This article does not contain any studies with human participants or animals.

Appendix

A sample policy specification with four permit rules (projected on critical functionalities).

```
<?xml version="1.0" encoding="UTF-8"?>
<Policy xmlns="urn:oasis:names:tc:xacml:3.0
    :core:schema:wd-17" PolicyId="
    urn:infobeyondtech:securitypolicytool:smartHome
    .spt:ABAC:CriticalPermitsPolicy"
    RuleCombiningAlgId="urn:oasis:names:tc:xacml:3
    .0:rule-combining-algorithm:ordered-deny-
    overrides" Version="1.0">
  <Target></Target>
  <Rule Effect="Permit" RuleId="rule_1">
    <Target>
      <AnyOf>
        <AllOf>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Spouse</
                AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::subjectcategory:accesssubject"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :subject:Role" DataType="http://www
                .w3.org/2001/XMLSchema#string"
                MustBePresent="true"></
                AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Xiaomi
                Mi Home Security Camera</
                AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:resource"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :resource:SecurityCamera" DataType=
                "http://www.w3.org/2001/XMLSchema#
                string" MustBePresent="true"></
                AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Camera
                ON/OFF</AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:action"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :action:Critical Functionalities"
                DataType="http://www.w3.org/2001/
                XMLSchema#string" MustBePresent="
                true"></AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Strong
                FMR</AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:environment"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :environment:Access Decision Score"
                 DataType="http://www.w3.org/2001/
                XMLSchema#string" MustBePresent="
                true"></AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
```

```
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">At Home<
                /AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:environment"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :environment:Location" DataType="
                http://www.w3.org/2001/XMLSchema#
                string" MustBePresent="true"></
                AttributeDesignator>
          </Match>
        </AllOf>
      </AnyOf>
    </Target>
  </Rule>
  <Rule Effect="Permit" RuleId="rule_2">
    <Target>
      <AnyOf>
        <AllOf>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Teenage<
                /AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::subjectcategory:accesssubject"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :subject:Role" DataType="http://www
                .w3.org/2001/XMLSchema#string"
                MustBePresent="true"></
                AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Xiaomi
                Mi Home Security Camera</
                AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:resource"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :resource:SecurityCamera" DataType=
                "http://www.w3.org/2001/XMLSchema#
                string" MustBePresent="true"></
                AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Camera
                ON/OFF</AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:action"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :action:Critical Functionalities"
                DataType="http://www.w3.org/2001/
                XMLSchema#string" MustBePresent="
                true"></AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Strong
                FMR</AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:environment"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :environment:Access Decision Score"
                 DataType="http://www.w3.org/2001/
                XMLSchema#string" MustBePresent="
                true"></AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">At Home<
                /AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:environment"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :environment:Location" DataType="
                http://www.w3.org/2001/XMLSchema#
                string" MustBePresent="true"></
                AttributeDesignator>
          </Match>
        </AllOf>
      </AnyOf>
    </Target>
  </Rule>
  <Rule Effect="Permit" RuleId="rule_3">
    <Target>
      <AnyOf>
        <AllOf>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Teenage<
                /AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::subjectcategory:accesssubject"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :subject:Role" DataType="http://www
                .w3.org/2001/XMLSchema#string"
                MustBePresent="true"></
                AttributeDesignator>
          </Match>
          <Match MatchId="urn:oasis:names:tc:xacml:1
              .0:function:http://www.w3.org/2001/
              xmlschema#string-equal">
            <AttributeValue DataType="http://www.w3.
                org/2001/XMLSchema#string">Google
                Home</AttributeValue>
            <AttributeDesignator Category="
                urn:oasis:names:tc:xacml:3.0
                ::attributecategory:resource"
                AttributeId="
                urn:oasis:names:tc:xacml:1.0
                :resource:VoiceAssistant" DataType=
                "http://www.w3.org/2001/XMLSchema#
                string" MustBePresent="true"></
                AttributeDesignator>
          </Match>
```

```
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">Online
              Shopping</AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::attributecategory:action"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :action:Critical Functionalities"
              DataType="http://www.w3.org/2001/
              XMLSchema#string" MustBePresent="
              true"></AttributeDesignator>
        </Match>
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">At Home<
              /AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::attributecategory:environment"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :environment:Location" DataType="
              http://www.w3.org/2001/XMLSchema#
              string" MustBePresent="true"></
              AttributeDesignator>
        </Match>
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">Strong
              FMR</AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::attributecategory:environment"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :environment:Access Decision Score"
               DataType="http://www.w3.org/2001/
              XMLSchema#string" MustBePresent="
              true"></AttributeDesignator>
        </Match>
      </AllOf>
    </AnyOf>
  </Target>
</Rule>
<Rule Effect="Permit" RuleId="rule_4">
  <Target>
    <AnyOf>
      <AllOf>
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">Spouse</
              AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::subjectcategory:accesssubject"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :subject:Role" DataType="http://www
              .w3.org/2001/XMLSchema#string"
              MustBePresent="true"></
              AttributeDesignator>
        </Match>
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">Google
              Home</AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::attributecategory:resource"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :resource:VoiceAssistant" DataType=
              "http://www.w3.org/2001/XMLSchema#
              string" MustBePresent="true"></
              AttributeDesignator>
        </Match>
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">Online
              Shopping</AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::attributecategory:action"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :action:Critical Functionalities"
              DataType="http://www.w3.org/2001/
              XMLSchema#string" MustBePresent="
              true"></AttributeDesignator>
        </Match>
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">At Home<
              /AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::attributecategory:environment"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :environment:Location" DataType="
              http://www.w3.org/2001/XMLSchema#
              string" MustBePresent="true"></
              AttributeDesignator>
        </Match>
        <Match MatchId="urn:oasis:names:tc:xacml:1
            .0:function:http://www.w3.org/2001/
            xmlschema#string-equal">
          <AttributeValue DataType="http://www.w3.
              org/2001/XMLSchema#string">Strong
              FMR</AttributeValue>
          <AttributeDesignator Category="
              urn:oasis:names:tc:xacml:3.0
              ::attributecategory:environment"
              AttributeId="
              urn:oasis:names:tc:xacml:1.0
              :environment:Access Decision Score"
               DataType="http://www.w3.org/2001/
              XMLSchema#string" MustBePresent="
              true"></AttributeDesignator>
        </Match>
      </AllOf>
    </AnyOf>
  </Target>
</Rule>
</Policy>
```

References

1. Ravidas, S., Lekidis, A., Paci, F., Zannone, N.: Access control in Internet-of-Things: a survey. J. Netw. Comput. Appl. **144**, 79–101 (2019). https://doi.org/10.1016/j.jnca.2019.06.0171610.01065
2. Naik, S., Maral, V.: Cyber security—IoT. In: RTEICT 2017—2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings 2018-January, pp. 764–767 (2018). https://doi.org/10.1109/RTEICT.2017.8256700
3. Ogonji, M.M., Okeyo, G., Wafula, J.M.: A survey on privacy and security of Internet of Things. Comput. Sci. Rev. **38**, 100312 (2020). https://doi.org/10.1016/j.cosrev.2020.100312
4. Fremantle, P., Scott, P.: A survey of secure middleware for the internet of things. PeerJ Comput. Sci. (2017). https://doi.org/10.7717/peerj-cs.114
5. Lee, S., Kim, J., Lee, S., Tech, G., Kim, H., Kim, J.: FACT: Functionality-Centric Access Control System for IoT Programming Frameworks. In: SACMAT'17, pp. 43–54 (2017)
6. He, W., Padhi, R., Ofek, J., Golla, M., Dürmuth, M., Fernandes, E., Ur, B.: Rethinking Access Control and Authentication for the Home Internet of Things (IoT). Usenix Sec (2018). https://www.usenix.org/conference/usenixsecurity18/presentation/he
7. Borse, Y., Chawathe, A.: A survey on access control in cloud computing. Int. J. Comput. Trends Technol. **59**(2), 81–84 (2018). https://doi.org/10.14445/22312803/ijctt-v59p113
8. Tian, Y., Zhang, N., Lin, Y.H., Wang, X.F., Ur, B., Guo, X.Z., Tague, P.: Smartauth: user-centered authorization for the internet of things. In: Proceedings of the 26th USENIX Security Symposium, pp. 361–378 (2017)
9. Yu, T., Sekar, V., Seshan, S., Agarwal, Y., Xu, C.: Handling a trillion (unfixable) flaws on a billion devices. In: Proceedings of the 14th ACM Workshop on Hot Topics in Networks, HotNets-XIV 2015, pp. 1–7 (2015). https://doi.org/10.1145/2834050.2834095
10. Ouaddah, A., Mousannif, H., Abou Elkalam, A., Ait Ouahman, A.: Access control in the Internet of Things: big challenges and new opportunities. Comput. Netw. **112**, 237–262 (2017). https://doi.org/10.1016/j.comnet.2016.11.007
11. Adda, M., Abdelaziz, J., McHeick, H., Saad, R.: Toward an access control model for IOTCollab. Procedia Comput. Sci. **52**(1), 428–435 (2015). https://doi.org/10.1016/j.procs.2015.05.009
12. Ye, N., Zhu, Y., Wang, R.C., Malekian, R., Lin, Q.M.: An efficient authentication and access control scheme for perception layer of internet of things. Appl. Math. Inf. Sci. **8**(4), 1617–1624 (2014). https://doi.org/10.12785/amis/080416
13. Yalcinkaya, E., Maffei, A., Onori, M.: Application of attribute based access control model for industrial control systems. Int. J. Comput. Netw. Inf. Secur. **9**(2), 12–21 (2017). https://doi.org/10.5815/ijcnis.2017.02.02
14. Hu, V.C., Ferraiolo, D., Kuhn, R., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K.: Guide to Attribute Based Access Control (ABAC) Definition and Considerations. NIST Special Publication (2014). https://doi.org/10.6028/NIST.SP.800-162
15. Oasis: eXtensible Access Control Markup Language. OASIS Standard (January):154 (2013)
16. Rath, A.T., Colin, J.N.: Strengthening access control in case of compromised accounts in smart home. In: International Conference on Wireless and Mobile Computing, Networking and Communications 2017-October, pp. 1–8 (2017). https://doi.org/10.1109/WiMOB.2017.8115827
17. Rath, T.A., Colin, J.N.: Adaptive risk-aware access control model for Internet of Things. In: Proceedings—2017 International Workshop on Secure Internet of Things, SIoT 2017, pp. 40–49 (2018). https://doi.org/10.1109/SIoT.2017.00010
18. Dong, Y., Wan, K., Huang, X., Yue, Y.: Contexts-states-aware access control for Internet of Things. In: Proceedings of the 2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design, CSCWD 2018, pp. 271–276 (2018). https://doi.org/10.1109/CSCWD.2018.8465364
19. Bezawada, B., Haefner, K., Ray, I.: Securing home IoT environments with attribute-based access control. In: Proceedings of the 3rd ACM Workshop on Attribute-Based Access Control, Co-located with CODASPY 2018, pp. 43–53 (2018). https://doi.org/10.1145/3180457.3180464
20. Sun, K., Yin, L.: Attribute-role-based hybrid access control. In: APWeb 2014 Workshops (61100181), pp. 333–343 (2014). https://doi.org/10.1007/978-3-319-11119-3_31
21. Aghili, S.F., Sedaghat, M., Singelée, D., Gupta, M.: MLS-ABAC: efficient multi-level security attribute-based access control scheme. Future Gener. Comput. Syst. **131**(January), 75–90 (2022). https://doi.org/10.1016/j.future.2022.01.003
22. Song, L., Li, M., Zhu, Z., Yuan, P., He, Y.: Attribute-based access control using smart contracts for the Internet of Things. Procedia Comput. Sci. **174**(2019), 231–242 (2020). https://doi.org/10.1016/j.procs.2020.06.079
23. Cathey, G., Benson, J., Gupta, M., Sandhu, R.: Edge centric secure data sharing with digital twins in smart ecosystems. In: Proceedings—2021 3rd IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications, TPS-ISA 2021, pp. 70–79 (2021). https://doi.org/10.1109/TPSISA52974.2021.00008
24. Goyal, G., Liu, P., Sural, S.: Securing Smart Home IoT Systems with Attribute-Based Access Control, vol 1. Association for Computing Machinery (2022). https://doi.org/10.1145/3510547.3517920
25. Gupta, M., Sandhu, R.: Towards activity-centric access control for smart collaborative ecosystems, vol 1. Association for Computing Machinery (2021). https://doi.org/10.1145/3450569.3463559, arXiv:2102.11484
26. Mawla, T., Gupta, M., Sandhu, R.: BlueSky: Activity Control: A Vision for "active" Security Models for Smart Collaborative Systems, vol 1. Association for Computing Machinery (2022). https://doi.org/10:1145/3532105.3535017
27. Zeng, E., Roesner, F.: Understanding and improving security and privacy in multi-user smart homes: a design exploration and in-home user study. In: Proceedings of the 28th USENIX Security Symposium, pp. 159–176 (2019)
28. Burakgazi Bilgen, M., Bicakci, K.: Extending attribute-based access control model with authentication information for Internet of Things. In: 2020 International Conference on Information Security and Cryptology, ISCTURKEY 2020—Proceedings, pp. 48–55 (2020). https://doi.org/10.1109/ISCTURKEY51113.2020.9307964
29. How biometrics will have a big impact on IoT technology - NEC NZ. https://www.nec.co.nz/marketleadership/publications-media/how-biometrics-will-have-a-big-impact-on-iot-technology/
30. Sugrim, S., Liu, C., McLean, M., Lindqvist, J.: Robust Performance Metrics for Authentication Systems. Network and Distributed Systems Security (NDSS) Symposium 2019 (February) (2019). https://doi.org/10.14722/ndss.2019.23351
31. Dunstone, T., Yager, N.: Biometric system and data analysis design, evaluation, and data mining. Springer US, (2009). 14:40. https://doi.org/10.1007/978-0-387-77627-92022-12-10
32. Dhir, V., Singh, A., Kumar, R., Singh, G.: Biometric recognition: a modern era for security. Int. J. Eng. Sci. Technol. **2**(8), 3364–3380 (2010)
33. Https://biolabcsruniboit/FvcOnGoing/UI/Form/PublishedAlgsaspx (2022) FVC-onGoing. https://biolab.csr.unibo.it/FvcOnGoing/UI/Form/PublishedAlgs.aspx#&&opq9kKfNMmetNyGfkPzA5od/P/tuPosw2DR8xqBRrz6hauX5tMGdzgwPF/egYeXkNIbFbE31Oxg

FjPHfByvIrbTvKn9EiOXZgtaXs7W2HpEj4EOZyEo0fs4RzbQGiqLmPbECcldIg/yR4Jl4iG4mhH2n7Uo37vRKR/RCw8F9HvRLJE+o

34. Dorizzi, B., Cappelli, R., Ferrara, M., Maio, D., Maltoni, D., Houmani, N., Garcia-Salicetti, S., Mayoue, A.: Fingerprint and on-line signature verification competitions at ICB 2009. In: Proceedings International Conference on Biometrics (ICB) 5558 LNCS, pp. 725–732 (2009). https://doi.org/10.1007/978-3-642-01793-3_74
35. Maio, D., Maltoni, D., Cappelli, R., Wayman, J.L., Jain, A.K.: FVC2002: second fingerprint verification competition. In: Proceedings—International Conference on Pattern Recognition, Vol. 16, No. 3, pp. 811–814 (2002). https://doi.org/10.1109/icpr.2002.1048144
36. Ameer, S., Sandhu, R.: The HABAC model for smart home IoT and comparison to EGRBAC. In: SAT-CPS 2021—Proceedings of the 2021 ACM Workshop on Secure and Trustworthy Cyber-Physical Systems, pp. 39–48 (2021). https://doi.org/10.1145/3445969.3450428