Extending Attribute-Based Access Control Model with Authentication Information for Internet of Things

Melike Burakgazi Bilgen
TOBB University of Economics and Technology
Ankara, Turkey
mbilgen@etu.edu.tr

Kemal Bicakci
TOBB University of Economics and Technology
Ankara, Turkey
bicakci@etu.edu.tr

***Abstract*—Internet of Things (IoT) brings not only wide range of opportunities but also security and privacy concerns. Consisting of many connected devices used in a highly interactive way, one of the main security concerns in IoT is unauthorized access. Traditional access control models do not support dynamic and fine-grained access control policies. Attribute-Based Access Control (ABAC) model is usually considered the most satisfactory access control model for running IoT applications. In this paper, we propose to take into the user authentication matching score obtained from a biometric authentication system consideration during making access control decisions. We emphasize the need of fine-grained access control and suggest to create access control policies per functionality of the device instead of per device regarding to the least privilege principle of information security. We give full or partial permission to certain functionalities of the IoT devices based on the user's authentication matching score thus provide more fine-grained and powerful access control mechanism. We present an extended Attribute-Based Access Control model that includes the assurance level of user authentication in access control policies and partially or fully permit the request accordingly.**

I. Introduction

Not so long ago, only computers, cell phones, and tablets were connected to the Internet. Today, with the dissemination of affordable new technologies, security cameras, microwaves, ovens, cars, and other industrial equipments have also become connected to the Internet. Internet of Things renders these equipments to computational devices by adding CPU and sensors and enables them to be connected to their users and other devices [1]. In this way, a huge network that consists of connected devices has been formed; 6 billion devices today and 20 billion devices in a few years [2].

This new technology brings not only wide range of opportunities but also many security and privacy concerns. As a result, some people will attempt to avoid adapting this technology in their daily lives [3]. Therefore, addressing security and privacy issues in IoT environments is one of the most essential needs.

In this paper, we focus on access control aspects of IoT to control and protect IoT devices from being used by unauthorized potentially malicious users. After examining current access control models, we decide that the Attribute-Based Access Control (ABAC) is the most suitable model for IoT. We extend the ABAC model by introducing the use of assurance level of user authentication in access control policies. The authentication matching score of the user is usually considered as either 0 or 1. However, for a fine-grained access control model, to determine the assurance level of user authentication there should be an authentication matching score for the role of the subject that has a value between 0 and 100. In our extended ABAC model, after users send an access request and get authenticated via one of the biometric authentications, they obtain an authentication matching score between 0-100. Then, the authentication matching scores of the users are saved and added to their request as another attribute. Then, partial or full permissions should be given for the requested functionalities of devices according to their authentication matching score. For instance, if the authentication matching score of the user for the babysitter role is 70 and above, an access for any camera related functionalities should be rejected. By this way, more secure and detailed access control can avoid the access of unauthorized and potentially malicious users. Even though there are studies that mention the importance of functionality-centric approach in IoT access control [4], [5], we have not found any study that use the authentication matching score and define permissions based on functionalities of the devices. We contribute to the literature by proposing our extended ABAC model which highlights the need of fine-grained access control mechanism and includes users' authentication matching scores in access policies for making decisions and giving partial permissions.

This paper is organized as follows. Section II provides background information about access control and discusses traditional access control models and ABAC in terms of IoT needs. Then it gives information about Extensible Access Control Markup Language (XACML) for ABAC implementations. It also provides further discussion for the importance of secure access control in IoT applications. Section III presents related work. Section IV describes the proposed extended ABAC model. Section V discusses the integration of the access

978-1-6654-1863-8/20/$31.00 ©2020 IEEE

control model with Azure IoT framework. Lastly, Section VI is our conclusion with some final remarks.

II. Background

Authentication, access control and authorization are the three main concepts of computer security.

Authentication is the process of verifying who you are. Access control is the mechanism that allows or denies user requests to resources based on the defined rules and policies. Access control rules and policies define which user accesses which resource with what permissions (R, W, RW). Access controls come after the authentication step and determine what the user is authorized to access. Finally, authorization is the process of permitting what authenticated users are allowed to access.

Access control mechanisms ensure the authorization in the resources [6]. The traditional access control methods are Discretionary Access Control (DAC), Mandatory Access Controls (MAC) and Role-Based Access Control (RBAC).

RBAC model, proposed by NIST in 1992, consists of users, roles, sessions, and authorization. In this model, access rules are defined based on the roles. Roles are assigned to users and permission is defined depending on their roles. This model is managed centrally [7]. Even though it is useful in small environments, it becomes unmanageable in big environments due to the high number of roles.

A. Attribute-Based Access Control (ABAC)

Despite the traditional access control models, ABAC does not define access permission between subjects and resources but uses subjects', resources' and environment's attributes. In this model, subjects' access requests on the resources are granted or denied depend on subject, resource and environment attributes [8].

Attributes consist of a name and a value. Subject attributes define the subject that makes the request to a resource. It can be age, role, department, management level, etc. They can be static like roles or dynamic like age and time. A resource is managed by ABAC rules and policies and its attributes are like resource type, resource feature, classification, etc. Finally, environment attributes can be about location, time and other dynamic attributes like threat level [7].

B. Extensible Access Control Markup Language (XACML) OASIS Standard

Extensible Access Control Markup Language (XACML) is an attribute-based access control standard. It defines a general way to express access control policies and request/response language. It also provides an architecture for computing and enforcing methods for access decisions [9]. NIST NGAC is another open standard supporting ABAC but we choose XACML for its strength in attribute and policy management to implement sample rules and policies for our model [10]. XACML structure consists of rules, policies, algorithms for rules and policies, and attributes of subject, object, action and environment. The policies are considered as logical conditions [10]. They may be very detailed and control access for a specific user on a specific object for a specific time frame. There are also general policies that apply for several users on several objects for more than one time frame. Flexibility of XACML is the most essential feature for designing access policies [11]. The architecture of XACML has the following flow with the responsible components [9]:

- Policy Enforcement Point (PEP): Meets a subject's request, sends it to Policy Decision Point (PDP) in order to make an authorization decision. According to the decision returned from PDP and obligations, it permits or denies the access to the object.
- Policy Decision Point (PDP): Evaluates the subject's request by finding corresponding access policies and makes authorization decisions. Also, if additional attributes are needed, it requests more attributes about the request from PIP.
- Policy Information Point (PIP): Returns the requested attributes that are missing in the original request to PDP. It is the source of the attributes.
- Policy Administration Point (PAP): Manages authorization policies and makes them available to PDP.

The logic behind XACML is based on attributes. The request of subjects consists of attributes of the subject, object, action and environment for the associated access. When a user makes a request for an operation on a related object, PEP meets the request and forwards to PDP for an authorization decision. PDP evaluates the attributes in the request by comparing the attribute values in the corresponding policies and makes an authorization decision. Then, PDP forwards this decision to PEP [9].

C. Internet of Things (IoT) and Access Control Models

Internet of Things can be considered as a combination of cyber-physical systems. It includes a wide range of devices/equipment that vary from energy stations, transportation services, financial systems, smart city infrastructures, smart vehicles to smart door locks, thermostats, security cameras, water sensors, motion sensors and health monitors [1]. A compromised IoT device might not only affect the application it uses, but also other cross devices that is dependent. If there is any misoperation, any unauthorized or malicious access or cyber-attack on IoT devices, it might not affect just a computer or a system, but it might have a huge effect on the physical world [12]. Therefore, cyber security and information security should be considered in detail in terms of IoT.

IoT makes our lives more comfortable by providing advanced and easily manageable systems, but the security vulnerabilities in IoT devices may cause customer dissatisfaction, violation of privacy, financial loss and even loss of life (in case attackers gain control of smart vehicles) [13]. For this reason, it is very essential to secure this technology transformation properly [14].

One of the main security concerns in IoT is unauthorized access control. The case that an unauthorized or a malicious user gains access to IoT devices and uses them for their

purposes by involving them in malicious network is definitely an undesirable condition [2]. Therefore, authentication and access control concepts are the key factors addressing security and privacy issues in IoT [15].

Access control models should take security, privacy and functionality into consideration [5]. Firstly, they should not define permissions based on the device, but on the functionalities of the device. Access permissions for each one of the capabilities of the device should be created so that a user accesses to only required features of the devices regarding to "least privilege" principle of information security. Therefore, IoT needs a fine-grained access control model. In ABAC model, it is possible to define fine-grained access control rules and policies in contrast to traditional access control models.

Also, IoT is a dynamic environment thus needs dynamic access control rules. ABAC method makes a decision depending on not only the roles of the subjects, like RBAC, but also the attributes of resources, environments and users. For more dynamic and complicated cases, instead of adding more roles and causing role explosion as in the RBAC method, more logical conditions are added to the rules and policies in the ABAC method [16]. With the help of environmental conditions such as location, date and time, threat levels and IP address, ABAC model is able to provide dynamic access control rules [17].

ABAC model has some advantages in terms of IoT. First of all, since the ABAC model provides interoperability, it makes it easier to work in collaborative environments. As long as the attributes of an unknown user meet the criteria of some of the existing policies and rules, the user gains access to the related source [15].

Secondly, in the ABAC model, access is granted if all criteria in related policies and rules are met, otherwise, it is rejected. Access policies and rules consist of different variations of subject, object and environment attribute values. Consequently, the number of different access condition combinations is much more than the traditional access controls offer [17]. This makes ABAC a fine-grained access control model [15].

In addition, as discussed in NIST ABAC guide [7] environment attributes are independent of subjects or objects and maybe about current time, location, and threat level. Yalcinkaya et al. [17] illustrate how the ABAC model can interpret different threat levels as environment or subject attributes. If the risk level is above some threshold, it can be automatically rejected based on the defined policies and rules. For instance, a threshold is determined as 5 by an existing access policy. If the user is in the enterprise network and authenticated by one-factor authentication method, the risk score is evaluated as 3 and the user gains access. While if the user is remote and authenticated by one-factor authentication method, the risk score is calculated as 7, then the user gets rejected. In case the same remote user is validated by multi-factor authentication method, the risk score is decreased to 4 and the user gains access [17]. With the ABAC model, it is possible to implement such scenarios by using environment attributes.

With all these advantages, ABAC makes it easier to implement real-life scenarios and meet more business requirements of IoT [7].

III. RELATED WORK

There have been several studies about access control models in IoT. Some researches propose ABAC and others extend ABAC by adding further features. In the first section, we mention the studies that propose ABAC for IoT. In the second section, we talk about the studies that extend ABAC for IoT. Finally, in the last section we gather the studies that use some sort of access control mechanisms or ideas that can be applied to our study.

A. Studies Proposing ABAC Model for IoT

Yalcinkaya et al. [17] mention the importance of access control systems and evaluate traditional access control models. They find the ABAC model the most suitable one for Information Control Systems (ICT). After explaining ABAC's structure and components well, they propose ABAC for ICT for its ability to provide fine-grained access control and having centrally administered access control policy mechanism. ICT is similar to IoT in terms of structure and the need of access controls, thus adapting ABAC to ICT as Yalcinkaya et al. [17] try to achieve gives an idea about how to adapt ABAC to IoT.

B. Bezawada et al. [14] propose the use of ABAC model for IoT access control. They list the security challenges in home IoT and choose NIST Next Generation Access Control (NGAC) specification for ABAC. They implement ABAC with NIST NGAC for home IoT by enforcing policies at the network level. They integrate NIST NGAC at the switch level and enforce policies at the flow level and packet level. They provide a good explanation about how to integrate ABAC with NIST NGAC at the network level. However, they think of access controlling for whole device without mentioning the need of access control policies for the functionalities of the devices. Also, they mostly focus on network-level access control enforcement without mentioning user authentication and access.

Sun and Yin et al. [18] recommend the Attribute-Role-Based Hybrid Model (ARBHAC) by blending ABAC and RBAC. They aim to solve RBAC's incapability in large-scale dynamic environments by blending it with ABAC. At the same time, they find ABAC complex in terms of policy management and permission assignment and try to ease it by blending with RBAC's role management. In RBAC, users have roles and each role is assigned to different permissions. They believe that although multiple attributes of the user can apply to a number of rules, only one role can be obtained. They add role-based permissions as another entity in policy management to simplify the complexity of policies. However, this model causes policy redundancy and conflict. Adding permission to users is simple in RBAC, but adding new roles requires a whole system to be updated. When the roles and permissions

change, it brings another administrator workload which is not easy to manage and track.

B. Studies Extending ABAC Model for IoT

A. T. Rath and J.N. Colin [3] extend ABAC model by taking an authentication system and an access control model together into consideration. They use association rule learning of UBA to create the user baseline by analysing the user's access logs. Then, they use this baseline to detect usage anomalies. They include the probability of certainty of the user in access policies as a user's behaviour analysis variable. This comes from the result of the UBA analysis of the authenticated user. They include the user's behavior attribute as a separate entity from environment attributes. Rath and Colin [3] define several policies for different thresholds by ensuring if the user is legitimate and requires extra security actions like sending security questions or notifying users, but they do not offer different functionalities for different thresholds. They only focus on if the user is malicious or legitimate based on the user's behaviour attribute.

In another study presented by Rath and Colin [19], they offer adaptive risk-aware access control and combine it with other access control systems such as ABAC. In risk-aware access control model, the risk value of the request is evaluated by using some techniques like machine learning and user's access request is reevaluated with the estimated risk value even though the user is already authenticated. Risk-aware information is external information that comes from different sources such as access history or other dataset used for estimating the risk associated with the request. In their model, they consider risk-aware information as different from environment attributes. Therefore, they contain risk-aware information and risk-estimation engine as separate entities in their model. They calculate risk-value by using risk-estimation engine based on risk-estimation information and function. Risk-estimation engine is thought as a component of Policy Decision Point. Finally, they use risk-avoidance enforcement module for enforcing some actions to avoid or minimise the risk. As an example of enforcing actions, requiring users to prove their identity by answering a security question or using other credentials are shown. For calculating risk-value, they apply Association Rule Learning technique on user access logs, get user access pattern and estimate risk-value for the access request. They use this value in ABAC rules and policies. They study continuous authorization and continuous decision but they do not consider granularity needs of IoT access control.

C. Other Studies

In IoT, access controls need to be based on the features of the devices instead of the whole device. In the study performed by W. He et al. [5], for each capability of the devices, 450 participants were asked by whom under which conditions (time, location, etc.) they prefer the related capability to be used. As a result of the study, it is observed that participants' answers vary according to the role of the user, capability of the device, time of the day, age of the users and other environmental conditions. They show how access decisions change for different roles/relationships on different capabilities of the devices in detail. They question on which contextual factors access control policies depend. Finally, they study on the consequences of wrong access decisions. For instance, Amazon Echo digital voice assistant device has several features such as playing music and online shopping. In this device, which feature can be used by which user should be clearly defined so that a guest or a child in the home does not perform online shopping via this device. W. He et al. [5] define the need for fine-grained access control in IoT well and provide various home IoT scenarios. They mention the need for more granular access control based on functionalities instead of per-device granularity. Even though they discuss access control and authentication requirements of home IoT in-depth and focus on the importance of access controls based on the functionalities rather than whole device, they do not relate the proposed access control specification to any access control model.

Another study that emphasizes the importance of access control based on functionalities of devices rather than per-device in IoT is presented by S.Lee et al. [4]. They propose a functionality-centric access control (FACT) framework which separates each functionality of the devices and defines access control policies for each of them. By this way, they increase the availability of the system since any interruption in one functionality would not affect other functionalities. They also try to achieve fine-grained access control by only giving an access to the needed functionalities and thus grant a subject the least privilege. This work illustrates the importance of the functionality-based access control in IoT well.

Table 1 summarizes the related work with regard to IoT access control requirements mentioned above.

TABLE I: IoT Access Control Requirements Survey

Ref No	Access Control Method	Assurance Level of User Authentication	Granularity
[3]	ABAC	Yes	No
[17]	ABAC	No	Yes
[18]	Attribute-Role Based Hybrid	No	Yes
[14]	ABAC	No	No
[5]	Not Specified	No	Yes
[20]	ABAC with Contexts-States-Awareness	No	No
[19]	Adaptive Risk-Aware ABAC	Yes	No
[21]	Not Specified	No	No
[4]	Functionality-centric Access Control System)	No	Yes
OUR MODEL	Extended ABAC	Yes	Yes

IV. OUR CONTRIBUTION

As seen in related work, there have been several proposals in terms of access control in IoT. Some of them cover the need of fine-grained access control model for IoT. None of the works

mentions the assurance level of user authentication comes from biometric authentication systems. Despite [3] takes into consideration the certainty of the user, it does not relate it to the authentication matching score of the user; instead, it uses behavior anomaly for getting certainty of the user. Even though some works only focus on authentication while others only concentrate on access control aspects, we believe authentication and access control should be both included in the access control model for IoT. As a result, an optimal access control model in IoT should cover assurance level of user authentication during access control decisions and grant fine-grained permissions for device functionalities. It is expected that due to IoT's high technology structure and nature, physical and behavioral biometric authentication methods will replace traditional authentication methods. Physical biometrics are about measurements of human body. It can be face shape, hand geometry, fingerprint and etc. Behavioral biometrics are associated with unique movements or habits of the users. Keystroke dynamics, speech patterns and signatures are examples of behavioral biometrics. Usually the first step is user authentication. An implicit assumption is the user is authenticated or not authenticated. On the other hand, biometric authentication systems calculate authentication matching scores of the user by performing the three steps, preprocessing, scoring and thresholding and makes authentication decision based on it [22]. If the user's authentication matching score is above the threshold, access is granted or otherwise rejected.

Without loss of generality, we consider an authentication matching score of the user has a value between 0 and 100. We assume a user will get different permissions for functionalities of the devices in cases where the user's authentication matching score is like 50 percent or 90 percent. Our goal is to make it possible to define more flexible and fine-grained access control policies based on the different authentication matching scores a user gets.

In our model, an authentication score for the subject is included as another environment attribute having a value between 0 and 100. It is used in access control policies in order to grant users full or partial permissions for related resources. For example, if the authentication matching score of the user for the teenager role is 60, he cannot perform online shopping but can play music via Amazon Echo device.

After the user gets authenticated via biometric authentication methods, we obtain and store the user authentication matching score. Then, we feed it to Policy Decision Point (PDP) so that PDP considers it during access decisions. Our extended version of ABAC architecture is presented in Figure 1.

In our extended ABAC model, a user sends an access request and gets authenticated via one of the biometric authentications. The user gets an authentication matching score between 0 and 100. Then, the authentication matching score of the user is saved and added to the request as another environment attribute. This is different from classical ABAC model. In classical ABAC, the user gets authenticated or not and an authentication matching score is not considered during

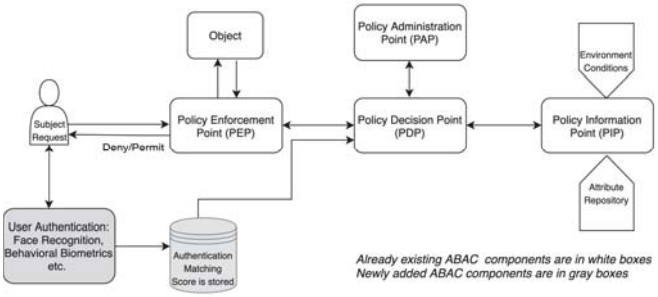

Fig. 1: Our Extended ABAC Model

access decision. Then, in our model Policy Enforcement Point(PEP) meets the access request and forwards it to Policy Decision Point for an authorization decision. The authentication matching score is also forwarded to Policy Decision Point. Policy Decision Point (PDP) takes into consideration attributes in the request including the authentication matching score during access decision. Also, PDP gets the needed attribute information and evaluates the request according to corresponding access rules and policies from Policy Administration Point. In classical ABAC, there is no partial permission. However, in our model PDP gives partial permissions based on the user authentication matching score. Then, PDP gives an authorization decision and sends it to PEP. PEP grants, partially grants or denies the access of the subject based on obligations and the decision PDP has made.

Inspired by the study presented at [5] we assume that we have a small home IoT environment consisting of cameras, smart lights, smart locks and voice assistants. We can perform actions like updating software, playing music, online shopping, turning the camera on/off, turning lights on/off, changing the angle of the camera and opening/closing door. We also assume roles of the users might be a spouse, teenage, child, babysitter and visiting family member. Finally, time, location of the user, age of the user, status of the device are the attributes we consider.

To illustrate how our model works, we implemented our small home IoT scenario in an XACML editor. The details of subject, object and environment attributes are shown in Figure 2. We add the authentication matching score as an environment attribute to our model, so it will be evaluated just as another attribute. The sample scenario is creating an access control policy for the online shopping feature of Amazon Echo Voice Assistant device. If its value is 90 percent and above for the requesting user, it will meet the rule's and policy's requirements to gain access. So, in the case the role of the subject is either Spouse or Teenage, the authentication score is 90 percent and above and the location is home, then the access is granted, and the access is rejected otherwise. Since we consider the authentication matching score as an environment attribute, we need to create policies for each situation. The code below is an example of including an authentication score in XACML Policy as an environment attribute.

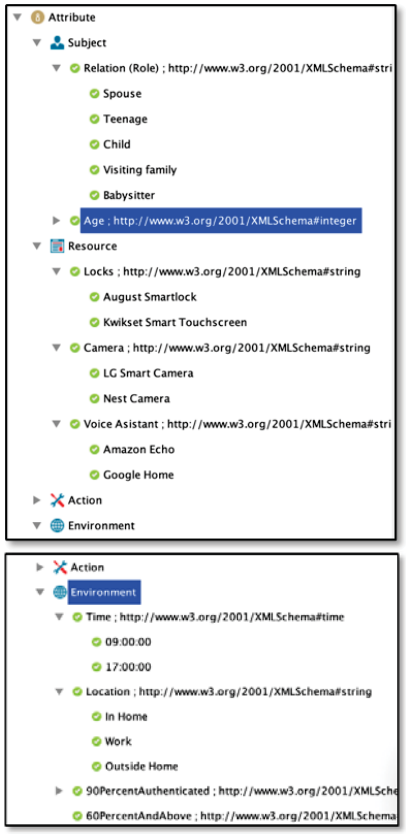

Fig. 2: Details of the Attributes in Small Home IoT

```
<Match MatchId="urn:oasis:names:tc:xacml:1.0
    :function:http://www.w3.org/2001/xmlschema#
    boolean-equal">
<AttributeValue DataType="http://www.w3.org/2001/
    XMLSchema#boolean">True</AttributeValue>
<AttributeDesignator Category="
    urn:oasis:names:tc:xacml:3.0
    ::attributecategory:environment"
 AttributeId="urn:oasis:names:tc:xacml:1.0
    :environment:90PercentAuthenticated" DataType="
    http://www.w3.org/2001/XMLSchema#boolean"
    MustBePresent="true">
</AttributeDesignator>
</Match>
```

Another access control policy for the online shopping feature of Amazon Echo Voice Assistant device is if the authentication score of the user is below 90 percent, the access is rejected regardless of other attributes of the request.

Moreover, Amazon Echo Voice Assistant device has music playing feature. Since playing music is not as a critical feature as online shopping, the access control policy can be more flexible. It can be like if the authentication score is 60 percent and above, location is home and subject is Spouse, Teenage, Child, Visiting Family or Babysitter, then the access is granted, and the access is rejected otherwise.

In another scenario, changing the camera's angle or turning the camera on/off can be serious actions that need to be managed carefully. If the user is Babysitter, any access to the camera should be forbidden. Thus, the access control policy for this case can be if the role of the subject is Babysitter and a request is made for the camera resource, the access is rejected regardless of other attributes in the request. On the other hand, if the time is after 6 pm and the location is home and the subject's role is Spouse and authentication score is 90 and above, then the access for turning the camera off is granted.

As the last scenario, we consider the locks and lights as resources. It should not be a big issue if the child turns on/off the lights, but it might be an issue if the child unlocks the door by himself. An access control policy for lights can be if the authentication score is 60 and above and the role is any and the location is home and time is after 5 pm, the access for turning on lights is granted. Lastly, the access control for locks is if the role is Child, the authentication score is 70 and above and the location is home, the access is rejected.

A. Authentication and Authorization

Home IoT users generally tend to give access control decisions based on both the role of the subject and the capability/functionality of the device. For instance, when an unauthorized family member accesses to camera logs may not be a serious issue while when a babysitter accesses to camera logs may cause serious problems. Therefore, the impact of false authorization decisions depends on the subject's role and granted device functionality [5]. It is vital to guarantee that the person granted an access is the one who is meant to be allowed in order to prevent the device from being used maliciously. The other critical issue is identifying the situation where the functionality is used in an unusual way [5]. In our model, determining the threshold value for an authentication score is essential to ensure the identity of the subject. More than one threshold value is needed since various access policies are required for building a more fine-grained access control model. For example,

- 90 and above will be one threshold for a subject to access full functionalities of the object,
- a value among 80 and 90 will be another threshold for accessing critical functionalities of the object,
- a threshold between 60 and 80 will be for accessing important capabilities,
- a value between 50 and 60 will be another one for accessing basic functionalities
- a value under 50 will be for denying access for all capabilities.

The values of the threshold define the power of the access control decisions, so it is significant to choose the reasonable threshold values for the rules and policies. Sugrim et al. [22] emphasizes the importance of the threshold value when explaining the three major operations, preprocessing, scoring and thresholding, of machine learning mechanism in authentication systems. After scoring operation, if the score is larger than the threshold, the access is granted. While, if the score is smaller than the threshold, the access is denied.

Therefore, choosing a good threshold value plays a critical role in terms of correctly authenticating the users. The main question is how to choose the optimum threshold value. Either user can determine the threshold values for each access policies manually or machine learning techniques including UBA can be applied. The first option may not be practical since there will be too many threshold values to determine for various scenarios. It will bring a heavy workload to users.

The second option needs detailed research but as one of the alternative ways, performance metrics can be used. We can include applicable performance metric types in access control policies. As stated in [22], the selection of a threshold shows a negotiation between error types. Even though eliminating error is not possible, it is doable to trade one error type to another. As a result, many threshold choices that make good compromise between error types will be available in case scoring function offers good separation. Performance metrics such as Equal Error Rate (EER) and Maximum Accuracy (ACC) fix a certain threshold and get a performance metric value from it. Usually, a threshold is chosen to optimize this metric. For example, Receiver Operator Characteristic Curve (ROC) is computed by varying the authentication threshold values from maximum to minimum possible values of the score and calculate True Positive Rate (TPR) and False Positive Rate (FPR) for each threshold value. Then EER, Area Under ROC Curve (AUROC) and Gini Coefficient (GC) performance metrics are used to summarize the computed ROC Curve [22]. In our case, we can write in the access control policy that if the performance metric is ROC curve and EER is below 0.4 for the given threshold, an access is granted. For instance, if the requested functionality is critical such as turning off the camera, the access control policy may be like if the time is after 6 pm and location is home and subject's role is Spouse and the EER corresponding the subject's authentication score is below 0.2, then the access for turning camera off is granted.

This method will relieve the user side as not requiring to define all threshold values for each scenarios. However, there will be cases that these performance metrics do not exist yet. In such cases, applied machine learning techniques can be used that is out of scope of our work.

V. IoT FRAMEWORK INTEGRATION

Extending ABAC is one side of this study and integrating our extended model to existing frameworks is the other side. After examining some IoT frameworks, we decided to proceed with Azure IoT Framework because of its practical installation and management.

Azure IoT uses Azure Active Directory for access control management. It provides easy management by taking advantage of RBAC. Role permissions are assigned to IoT devices. There are several built-in roles and also custom roles can be created. To implement our scenarios, we created custom roles corresponding to each threshold value we had defined in the previous section as in Figure 3.

We determined what these roles can perform by assigning permissions to these roles in detail as shown in Figure 4. For

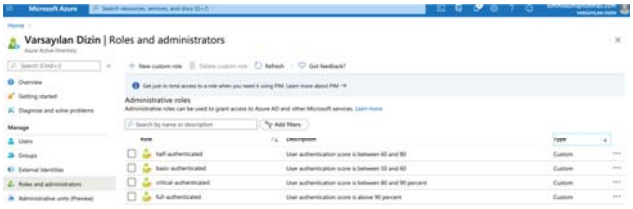

Fig. 3: Custom Roles in Azure Platform

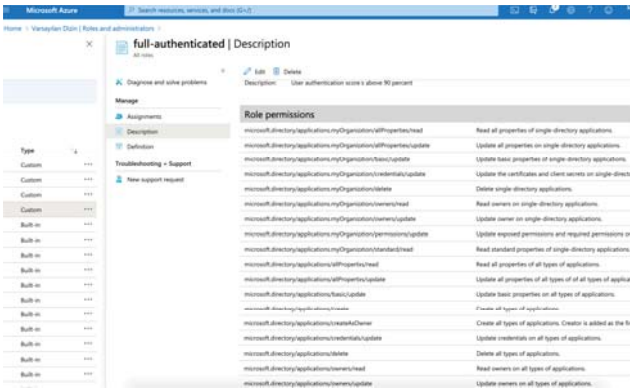

Fig. 4: Some Permissions of the Full Authenticated Role in Azure Platform

instance, in our case a user who has an authentication score 90 and above will have a full-authenticated role. The user can perform full action on the requested resource.

A user with authentication score 95 will be assigned to a full-authenticated role and will be allowed to perform create, update, read and delete permissions for the requested resource. While a user with authentication score 55 will be assigned to a basic-authenticated role and have limited read permissions for the requested resource. As far as we see, in Azure IoT permissions are assigned per device not per functionality of the devices. To be able to construct more fine-grained access control, there is a need for permission assignment at the functionality level. Azure IoT can be extended with this aspect. Also, it does not support dynamic access decisions since lack of attributes because of its underlying RBAC logic. Therefore, it seems a good idea to extend Azure IoT framework by adding an access control module considering the means of device functionalities and attributes. We leave it as a future study.

VI. CONCLUSION

IoT has a complex and dynamic environment due to a variety of devices and the diversity and instability of users. Therefore, it needs a dynamic, fine-grained and context-aware access control model. In this paper, we discussed access control models in Internet of Things and proposed our extended version of the ABAC model. We aim to extend the ABAC model to a more fine-grained access control model by including authentication scores in addition to subject, object and environment attributes in access control rules and policies to make more detailed and granular access decisions. We underlined the need of functionality based access decision

rather than device based. The point to consider about our model is its usability even if its necessity is clear. The question is whether the home IoT users find this approach usable and manageable. Also, we focused on the importance of threshold selection and related it with performance metrics. We also analyzed some IoT frameworks to practice how to integrate and implement our model in real IoT environment. Further studies need to be performed in terms of integration access control models and IoT framework and we leave it as a future work.

REFERENCES

[1] P. C. Van Oorschot, "Internet of things security: Is Anything New?" *IEEE Security and Privacy*, vol. 16, no. 5, pp. 3–5, 2018.

[2] S. Naik and V. Maral, "Cyber security - IoT," *RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings*, vol. 2018-Janua, pp. 764–767, 2018.

[3] A. T. Rath and J. N. Colin, "Strengthening access control in case of compromised accounts in smart home," *International Conference on Wireless and Mobile Computing, Networking and Communications*, vol. 2017-Octob, pp. 1–8, 2017.

[4] S. Lee, J. Kim, S. Lee, G. Tech, H. Kim, and J. Kim, "FACT : Functionality-centric Access Control System for IoT Programming Frameworks," *SACMAT'17*, pp. 43–54, 2017.

[5] W. He, R. Padhi, J. Ofek, M. Golla, M. Dürmuth, E. Fernandes, and B. Ur, "Rethinking Access Control and Authentication for the Home Internet of Things (IoT)," *Usenix Sec*, 2018. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/he

[6] N. Ye, Y. Zhu, R. C. Wang, R. Malekian, and Q. M. Lin, "An efficient authentication and access control scheme for perception layer of internet of things," *Applied Mathematics and Information Sciences*, vol. 8, no. 4, pp. 1617–1624, 2014.

[7] V. C. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, and K. Scarfone, "Guide to Attribute Based Access Control (ABAC) Definition and Considerations," 2014. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-162.pdf

[8] T. Priebe, W. Dobmeier, C. Schläger, and N. Kamprath, "Supporting attribute-based access control in authorization and authentication infrastructures with ontologies," *Journal of Software*, vol. 2, no. 1, pp. 27–38, 2007.

[9] Oasis, "eXtensible Access Control Markup Language," *OASIS Standard*, no. January, p. 154, 2013.

[10] D. Ferraiolo, R. Chandramouli, R. Kuhn, and V. Hu, "Extensible Access Control Markup Language (XACML) and Next Generation Access Control (NGAC)," in *2016 ACM International Workshop*. New York, USA: ACM Press, 2016, pp. 13–24. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2875491.2875496

[11] C. Dewi, P. Kencana, H. Riis, and D. Version, "Modelling and Analysing Access Control Policies in XACML 3.0 Carroline," 2015.

[12] T. Yu, V. Sekar, S. Seshan, Y. Agarwal, and C. Xu, "Handling a trillion (unfixable) flaws on a billion devices," *Proceedings of the 14th ACM Workshop on Hot Topics in Networks, HotNets-XIV 2015*, pp. 1–7, 2015.

[13] E. Fernandes, A. Rahmati, K. Eykholt, and A. Prakash, "Internet of Things Security Research: A Rehash of Old Ideas or New Intellectual Challenges?" *IEEE Security and Privacy*, vol. 15, no. 4, pp. 79–84, 2017.

[14] B. Bezawada, K. Haefner, and I. Ray, "Securing Home IoT Environments with Attribute-Based Access Control," *Proceedings of the 3rd ACM Workshop on Attribute-Based Access Control, Co-located with CODASPY 2018*, pp. 43–53, 2018.

[15] A. Ouaddah, H. Mousannif, A. Abou Elkalam, and A. Ait Ouahman, "Access control in the Internet of Things: Big challenges and new opportunities," *Computer Networks*, vol. 112, pp. 237–262, 2017. [Online]. Available: https://doi.org/10.1016/j.comnet.2016.11.007

[16] Y. Zhang and X. Wu, "Access Control in Internet of Things: A Survey," *DEStech Transactions on Engineering and Technology Research*, no. apetc, 2018.

[17] E. Yalcinkaya, A. Maffei, and M. Onori, "Application of Attribute Based Access Control Model for Industrial Control Systems," *International Journal of Computer Network and Information Security*, vol. 9, no. 2, pp. 12–21, 2017.

[18] K. Sun and L. Yin, "Attribute-Role-Based Hybrid Access Control," *APWeb 2014 Workshops*, no. 61100181, pp. 333–343, 2014.

[19] T. A. Rath and J. N. Colin, "Adaptive Risk-Aware Access Control Model for Internet of Things," in *Proceedings - 2017 International Workshop on Secure Internet of Things*. SIoT 2017 (2018), 2018, pp. 40–49.

[20] Y. Dong, K. Wan, X. Huang, and Y. Yue, "Contexts-States-Aware Access Control for Internet of Things," *Proceedings of the 2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design, CSCWD 2018*, pp. 271–276, 2018.

[21] N. Amraoui, A. Besrour, R. Ksantini, and B. Zouari, "Implicit and Continuous Authentication of Smart Home Users," *International Conference on Advanced Information Networking and Applications*, no. 1, pp. 834–845, 2019. [Online]. Available: http://link.springer.com/10.1007/978-3-030-15032-7

[22] S. Sugrim, C. Liu, M. McLean, and J. Lindqvist, "Robust Performance Metrics for Authentication Systems," *Network and Distributed Systems Security (NDSS) Symposium 2019*, no. February, 2019.