Uncertainty-Aware Authentication Model for Fog Computing in IoT

Mohammad Heydari
Department of Computing and Informatics Bournemouth University,
Bournemouth, UK
mheydari@bournemouth.ac.uk

Alexios Mylonas
Department of Computing and Informatics Bournemouth University,
Bournemouth, UK
amylonas@bournemouth.ac.uk

Vasilios Katos
Department of Computing and Informatics Bournemouth University,
Bournemouth, UK
vkatos@bournemouth.ac.uk

Emili Balaguer-Ballester
Department of Computing and Informatics Bournemouth University,
Bournemouth, UK
eb-ballester@bournemouth.ac.uk

Vahid Heydari Fami Tafreshi
School of Computing and Digital Technologies
Staffordshire University
Stoke, UK
v.heydari@staffs.ac.uk

Elhadj Benkhelifa
School of Computing and Digital Technologies
Staffordshire University
Stoke, UK
e.benkhelifa@staffs.ac.uk

Abstract— Since the term "Fog Computing" has been coined by Cisco Systems in 2012, security and privacy issues of this promising paradigm are still open challenges. Among various security challenges, Access Control is a crucial concern for all cloud computing-like systems (e.g. Fog computing, Mobile edge computing) in the IoT era. Therefore, assigning the precise level of access in such an inherently scalable, heterogeneous and dynamic environment is not easy to perform. This work defines the uncertainty challenge for authentication phase of the access control in fog computing because on one hand fog has a number of characteristics that amplify uncertainty in authentication and on the other hand applying traditional access control models does not result in a flexible and resilient solution. Therefore, we have proposed a novel prediction model based on the extension of Attribute Based Access Control (ABAC) model. Our data-driven model is able to handle uncertainty in authentication. It is also able to consider the mobility of mobile edge devices in order to handle authentication. In doing so, we have built our model using and comparing four supervised classification algorithms namely as Decision Tree, Naïve Bayes, Logistic Regression and Support Vector Machine. Our model can achieve authentication performance with 88.14% accuracy using Logistic Regression.

Index Terms— Uncertainty, Authentication, Fog Computing, Mobile Edge Computing, Internet of Things, Supervised Learning, Prediction Model

I. Introduction

Fog computing is considered as the extension of the cloud computing to the network edge in the context of Internet of Things (IoT) [1]. It introduces a new breed of computation and communication by extending the connectivity between a huge number of heterogenous, decentralized and dynamic devices without the intervention of third parties. Fog computing has a number of advantages like real-time access, location awareness, wireless access, heterogeneity and scalability which apart from the obvious opportunities, introduces great security and privacy challenges [2]. Among the various security challenges in IoT, authentication is a crucial and open challenge in fog computing [3], [4]. Moreover, fog inherent characteristics like *scalability*, *interoperability*, *dynamism* and *wireless access* exaggerate the security challenges that are related to the field of access control. *Dynamism* may result in uncertainty in authentication because persistent authentication in the fog environment does not provide robust security protection mechanism so the need for real-time tracking of the rapid changes is vital and its not easily achievable in the fog because fog nodes frequently join and leave the fog layer [2]. Furthermore, scalability can increase dynamism in a way that having complete information to make real-time access decision is imposible. Furthermore, network and service dependency in a *heterogeneous* environment like fog can cause delay in network delivery and this leads to uncertainty in making access decision because required information for a real-time access decision is delivered with delay. In all of the above cases the lack of information caused by the inability of tracking those changes results in uncertainty. In summary, uncertainty in authentication comes to play where an access decision needs to be made based on incomplete information. To address this challenge, we first define uncertainty in authentication by considering the "liklihood of an incident occuring" per each authentication request and then try to measure the uncertainty and build a data-driven model to handle uncertainty in authentication.

The rest of the paper is organized as follows. In section II background and related work are presented. In section III our proposed model is presented. Our methodology is thoroughly discussed in section IV. Section V consists of the results that come from the conducted experiments. It also discusses the results. Finally, section VI contains our conclusion and future work.

II. Background And Related Work

Access control is a mechanism by which system resources can be used only by authorized entities based on a policy. Access control consists of the following functions namely, Authentication, Authorization and Auditing [5]. In this research we focus on uncertainty aspect of the authentication phase of the access control.

978-1-7281-1796-6/19/$31.00 ©2019 IEEE

Authentication is one of the access control functions which is defined as a verification process to check whether the credentials of an entity is valid. An Access control system may have some of the following characteristics which are often used to evaluate the performance of the access control system [6]: delegation, revocation granularity, flexibility, scalability, lightweight, heterogeneity and context-aware. In order to build an access control model, the following specifications must be taken into considerations due to the distinct characteristics of fog computing: *i) Dynamism:* If the access decision must change, due to the changes in the environment attributes, while the access is granted, then, the access control system is classified as dynamic. Otherwise, if the changes do not affect the access decision, then the access control system is static. Considering dynamism in authentication for fog computing is important, due to the rapid changes of contextual parameters that occur in end-user devices. *ii) Scalability* in access control must be evaluated by three dimensions, namely an access controls has: *a) Subject/Object (entities) scalability* if increasing the number of entities does not lead to an overhead in processing time or workload, b) *Policy rules scalability:* if increasing the number of access rules does not lead to overhead in terms of processing time or workload., and *c) Extensibility* if it has the ability to extend its structure to cover more sub-systems and domains. The third form of scalability can be achieved through building de-centralized structure rather than centralized structure in scalable environments like fog computing. *iii) Heterogeneity/Interoperability:* In fog computing, entities have dependencies and their workflows are tightly convergent, which increases complexity. For this reason, any access control breach in such an environment can be more disruptive compared to traditional computing environment. Furthermore, as fog computing is composed of different platforms, enabling technologies and domains, designing an access control model to regulate access inter/intra domains or technologies is a must. *iv) Context-Aware:* It refers to the ability of the access control system to take contextual attributes to make an access decision. Considering contextual parameters in access decision brings flexibility in terms of tracking subject, object and environment changes if those changes have impacts on the decision.

The above evaluation criteria uncover limitations in the both traditional access control models like DAC, MAC, RBAC and emerging access control models like CapBAC, ABAC and making them inapplicable to any scalable, dynamic and heterogenous environment like fog computing. For this reason, a number of studies suggested new access control models as the extension of the above models to be deployed in the context of the cloud and fog computing.

M. H. Ibrahim [7] proposed an authentication scheme for fog computing. This scheme enables any fog user or node to mutually authenticate each other without third party intervention. The limitation of this work is that the scheme forces the fog nodes to store the credentials of all fog users in the same trust domain. P Hu et al. [8] proposed an authentication scheme using face identification. The proposed scheme consists of three parts namely, identity authentication, data integrity and data encryption to protect confidentiality, integrity and availability in fog computing. Using multiple encryption algorithms like AES and secure hash function like SHA-1 is the main drawback of the proposed scheme due to the limitations of resource constraint devices in fog and mobile edge computing. Dos Santos et al. [9] proposed a Risk Aware Access Control (RAAC) method for the cloud. In this method, if the subject of access is in the same cloud federation as the object, ABAC policies are enforced by the cloud service provider offering the object. otherwise, risk policies are evaluated against the attributes of the subject and access is granted only if the risk is below a determined threshold. Dos Santos et al. improved their approach in [10] and enriched their method by applying RAAC not only for intra-cloud access decisions, but also for inter-cloud access decision. Daniel Ricardo et al. [11] proposed a risk-aware framework to enforce RAAC policies in the cloud. This work is based on the extension of XACML and aggregates various risk factors to calculate the final value of the risk. Risk itself is measured based on the impact that access can cause.

A number of studies suggested resilient access control paradigms to deal with indeterminant data access scenarios. These paradigms include (i) Break-The-Glass Access Control (ii) Optimistic Access Control, and (iii) Risk-Aware Access Control [12], [13]. Ferreira [14] proposed a model called Break-The-Glass (BTG) to allow policy overrides. The aim of this model is to allow unanticipated access to be provided in unexpected situations. The main application of this method is in the emergency situations in the healthcare system [15]. One of the most important problems with the BTG is the scalability of policy overriding. By increasing the number of policy overriding in a scalable environment like IoT, the access monitoring and misuse detection become impossible [16]. In cases such as emergency healthcare services, the capability of an access control system to provide openness and availability is more necessary than confidentiality [17]. In this context, optimistic access control has been proposed, which assumes that most access requests will be legitimate. An optimistic approach permits the subject to exceed their normal access rights. Therefore, putting additional control layer to protect the asset from misuse is recommended for optimistic access control. This approach suffers from the lack of scalability in terms of policy rules. Risk-Aware access control was proposed to assess the risk of the authentication request to determine whether the access to a resource should be granted [18]. Nurse et al. [19] argue that by considering the IoT-related characteristics such as scalability, heterogeneity and dynamism, the current risk assessment approaches are inadequate for environments like IoT due to the (i) Limitation of periodic assessment such environments, (ii) Lack of knowledge about entities (i.e. Fog nodes) and (iii) Interoperability and dependency challenges.

III. Proposed Model

Uncertainty has not had the attention that deserves as a challenge in fog computing in the context of IoT, compared to other challenges that are well-studied in the relevant literature, such as scalability, heterogeneity, interoperability and dynamism [20], [21], [22], [23], [24]. However, as this work stresses, uncertainty should be considered when making an access control decision in the context of IoT. Otherwise, if the decision is based on deterministic rules regardless of the uncertainty concept, it does not fit in dynamic environment like fog computing. In this work we consider that uncertainty is caused by the lack of information about the likelihood of an incident occurring. Therefore, we define uncertainty in authentication as the incompleteness of information regarding the likelihood of whether the acceptance of an authentication request leads to an incident. For instance, assume that "Alice" attempts to authenticate to a system. It is supposed that authenticating her, endangers the system (the access exposes an asset to a threat) with a probability 60%. The closest concept to uncertainty is “Risk”. In one hand, risk is defined as a measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence [25]. On the other hand, uncertainty is defined as the lack of information about the likelihood of an event occurring. Therefore, “likelihood of event occurring” is common between these two concepts. For this reason, in this paper, uncertainty handling block is referred to ‘risk engine’ due to the such a resemblance. To handle uncertainty in authentication, we have proposed a data-driven model based on the extension of ABAC. Figure (2) depicts the architecture of our model based on XACML [26] . XACML is the standard and policy language for ABAC so we have built our architecture based on it. In our model, 1) users send authentication requests to the Policy Enforcement Point (PEP). PEP that is the interface between the system and the user sends the request to Policy Decision Point (PDP), which is responsible for gathering policy related to the specified resource from Policy Administration Point (PAP). 3) PDP requests policy from PAP. 4) PAP is responsible to provide requested policy to PDP. 5) PDP also requests subject, object and environment attributes related to the request from Policy Information Point (PIP). 6) PIP is responsible to gather attributes related to the request (subject, object, environment) and makes it available to PDP. 7) PDP sends the gathered information by PIP to Indeterminacy Estimation Point (IEP) and requests the risk engine to calculate the uncertainty values associated to the authentication request. 8) IEP sends request to risk engine to calculate the value of uncertainty associated with the authentication request. 9) Risk engine returns the calculated the overall value for the uncertainty. 10) IEP returns the value of uncertainty based to PDP. 11) PDP makes final access decision using related policy and the value of indeterminacy which was provided by IEP. Then the decision will be forwarded to PEP. 12) PEP fulfills the obligations based on the decision.

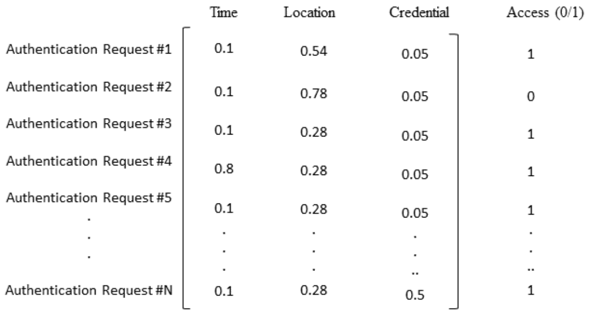
IV. Methodology

The output of an access control system must be classified into a binary decision: Access or Deny. For this reason, classification techniques need to be considered in order to build a data model. In this work we applied three different classification algorithms, namely Decision Tree, Random Forest and Logistic Regression. We also synthesized a dataset based on the state-of-the-art researches to develop our data-driven model with. In order to conduct our experiments, we have used MATLAB version ‘2017b’ to synthesize our dataset. We have also applied machine learning algorithms using scikit-learn package version 0.20.2 with python version 3.7.1.

A. Dataset Synthesis

A major challenge facing researches in the field of authentication is the lack of publicly available dataset that address our needs. Those datasets that are publicly available like LANL [27] and Bank Note [28] don’t consist of required features. Therefore, to the best of our knowledge, authentication dataset consisting of our required features are not publicly available. As a result, we synthesized an authentication dataset that enables the robust testing of our access control decision-making approach. Furthermore, our model is an extension of ABAC and we have considered the following attributes for each authentication request to generate corresponding uncertainty values that we call it as risk values: (i) Time of the request (ii) Location of the request and (iii) Credentials provided by the user.

Having synthesized the uncertainty values for these attributes, we will come up with the overall uncertainty value per user request using our risk engine. As depicted in Figure (1), PDP will receive the overall uncertainty value and make an authentication decision using pre-defined policies (PIP & PAP). The uncertainty values for each of these attributes will be represented by probability distribution functions (PDFs) because in real word scenario these attributes derived from stochastic processes so they should be represented using PDFs. In doing so, each attribute should be studied separately to determine the PDF that reflects the uncertainty in authentication by presenting the likelihood of the incident occurrence for a selected attribute. The outcome of the dataset synthesis process is an uncertainty matrix consisting of generated uncertainty values for these three attributes (Figure (1)). In the reminder of this section, the synthesis of values for each attribute is discussed separately.



	Time	Location	Credential	Access (0/1)
Authentication Request #1	0.1	0.54	0.05	1
Authentication Request #2	0.1	0.78	0.05	0
Authentication Request #3	0.1	0.28	0.05	1
Authentication Request #4	0.8	0.28	0.05	1
Authentication Request #5	0.1	0.28	0.05	1
.	.	.	.	.
.	.	.	.	.
.	.	.	..	..
Authentication Request #N	0.1	0.28	0.5	1

Figure (1): Uncertainty Matrix consisting of generated uncertainty values

1- Time

In order to determine the PDF for authentication request time we have taken the following considerations:

i) The pattern for the time of authentication request may vary from one case study to another. It follows the business model of the service in which the authentication process is embedded. For example, email services are deployed to be accessible 24 hours a day, 7 days a week and generally no restriction is defined for the sake of access to the email services in terms of time. In such a scenario, time of the authentication requests follows uniform distribution. On the contrary, if the access to a service is mostly demanded during a specific time period like work hours (e.g. 9AM to 5PM) then we should take those time preferences into consideration and find the corresponding PDFs.

ii) In this research, we have considered a case study composed of a company that authenticates its users in order to give them the access to its resources. We also assume that the majority of users send authentication requests during work hours (9AM-5PM) and the number of requests before 9AM and after 5PM plummeted gradually. In order to make the scenario more realistic, we also assume that the

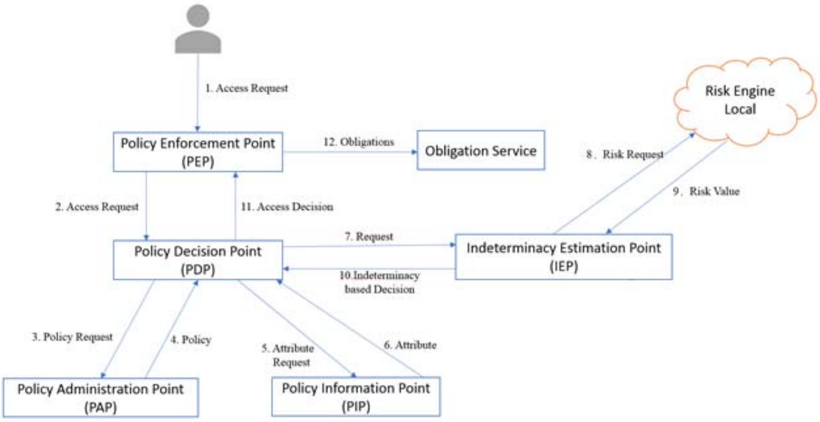


Figure (1):Architecture of the proposed model

number of requests between 12-13 decreases due to the rest/lunch time.

iii) Based on the above considerations, we have divided the time of the authentication requests into 11 time-slots. We assumed a weight in terms of probability for each time-slot that indicates the likelihood of making a request by the users. We have also defined an uncertainty value for each time-slot. The logic behind these values is that if an authentication request is made during work hours it is generally less prone to incident than any request which is made out of work hours. Therefore, the uncertainty value for any request that is made out of the work hours increases gradually. Furthermore, for all authentication requests during work hours the least value of risk was assigned due to the threat of insiders.

iv) Uncertainty values for the time of authentication request were generated using two PDFs. First, samples were randomly drawn from a multinomial distribution to determine the time-slots then a uniform distribution was similarly applied to randomly determine the time of the request within nominated time-slot. Finally, uncertainty values for the generated request times were assigned. Table 1 shows the information about time-slots and associated weights and uncertainty values that we defined based on the above discussed points in details.

Table 1: Defined time-slots and associated probabilities and risk values

Time-Slot	Weight (Probability)	Uncertainty Value
[1-5)	0.005	0.80
[5-7)	0.006	0.75
[7-8)	0.01	0.60
[8-9)	0.04	0.50
[9-12)	0.35	0.10
[12-13)	0.10	0.20
[13-17)	0.40	0.10
[17-18)	0.06	0.40
[18-19)	0.02	0.50
[19-23)	0.007	0.70
[23-1)	0.002	0.90

2- Location

One of the advantages of our proposed method is the capability of doing uncertainty analysis for mobile users, a fundamental requirement since the number of security and privacy incidents caused by them is rapidly increasing [29]. Towards this goal, changes in the location of mobile users need to be modelled first. In order to generate location- based uncertainty values we have taken the following points into considerations

i) A number of studies suggested normally distributed locations of mobile users in communication [30] [31], [32], . Thus, based on these results, we have used the Gaussian PDF to generate data for any specific location of the mobile user in a two-dimensional grid, (X: longitude and Y: latitude).

ii) We have defined a scenario consisting of three Point of Interests (PoI) to make this case study more challenging and realistic where the number of PoIs may vary from one case study to another. We have applied a mixture Gaussian PDF based on the mentioned studies to generate locations of authentication requests. According to these assumptions our PDF consists of three Gaussian factors in which each of them has a weight and each PDF belongs to one PoI respectively:

$$G_T = \alpha G_1 + \beta G_2 + \gamma G_3 \quad (1).$$

We expect that most of the authentication requests to be sent from or around the first PoI (which is generated using G_1) such that the magnitude of α coefficient was chosen in a way that reflects this fact. Next, the second PoI generates the second highest number of requests (using G_2) whist the third PoI should generate the smallest number of authentication request associated with location (using G_3) so that:

$$\alpha > \beta > \gamma \quad (2)$$

iii) The above formulation, which is based on Gaussian mixture model, provides a practical way to generate uncertainty values for the location attribute. For our dataset, we have generated 5000 authentication requests in terms of location (mobile and fixed) along with a map of area 2000m * 2000m, which contains three PoIs namely PoI_1, PoI_2 and PoI_3. Table (2) shows the assigned values as gaussian parameters μ and σ for our three gaussian factors, which were used to generate random values in both dimensions X and Y. Theses parameters were suggested based on the location of our three PoIs.

iv) Uncertainty values were defined for each PoI along with 5 different Uncertainty Areas (UAs). Figure (3) shows the UAs for PoIs. In order to define UAs for each PoI, five circles were drawn with the PoI point as the center and with (2n+1)*r as radius (n=0,1,2,3… and r=200m). The number of circles and the length of the radius may vary from one case study and thus is considered a system parameter.

v) Data for the authentication request for the location attribute were generated as follows: First, a multinomial PDF was applied to randomly choose a specific PoI from three PoIs using nominated weights (α, β,γ) as probabilities. Second, PoI associated Gaussian PDF applied to generate the X and Y points of the location. Third, according to the location of the generated point on the map an Uncertainty Value (UV) was generated using the following formula:

$UV_{Total}= \alpha$ *(UV assigned by PoI_1) + β * (UV assigned by PoI_2) + γ*(UV assigned by PoI_3) (3)

According to the concept of the mixture model, in order to calculate the total UV of any given location the UV assigned by all PoIs should be considered.

The value assigned by each given PoI in the above formula depends on the UAs in which the point has fallen.

Table (2): Assigned values for Gaussian PDFs parameters

PoI_1	PoI_2	PoI_3
μ_x=200 , σ_x=100	μ_x=1000 , σ_x=500	μ_x=1400 , σ_x=800
μ_y=200 , σ_y=100	μ_y=600 , σ_y=400	μ_y=1400 , σ_y=800
α= 0.65	β= 0.20	γ= 0.15

3- Credential

The most usual form of authentication is using username and password. We have considered this information as the credential for this research. In order to generate risk values for the credential we have taken the following points into considerations:

i) Usernames and passwords entered by users makes three possibilities: (i) both username and password provided by the user are correct (ii) only the username is correct and (iii) only the username is incorrect. Data for the three possible states was generated form a multinomial PDF as described below.

ii) Generally, most users enter username and password correctly. Otherwise, most users enter the username correctly but enter the password incorrectly. These were considered when assigning probability values and associated uncertainty values (UV) listed in Table 3 for these three states.

Table (3): Assigned values for credential associated PDF and corresponding risk values

User & Password are correct	User is correct but Pass	User & Password are incorrect
Probability: 0.85	Probability: 0.10	Probability: 0.05
UV: 0.05	UV: 0.70	UV: 0.95

4- Access Decision

After generating the uncertainty values for each attribute in the uncertainty matrix shown in Figure (2) the final uncertainty value is calculated for each request in order to make an authentication decision. The final value for each authentication request was calculated by averaging the uncertainty values of time, location and credential. Generally, credential is the most important authentication attribute in comparison with time and location. We have added weights to the generated uncertainty values to show the priority and importance of the attributes. The magnitude of these weights may vary based on the research priorities. Therefore, we have calculated the weighted arithmetic mean by averaging of weighted risk values (weight values: Time=2, Location=3 and Credential=5). Finally, for labeling the dataset we have used the final uncertainty value for each request as the probability for binomial distribution to determine the class of the result: {0:Deny and 1:Access}.

B. Prediction Models for Authentication

In this research we have applied four supervised classification algorithms to build our prediction models. Further classification algorithms like KNN, ANN and Random Forest will be applied in the future work.

1- Decision Tree

Decision tree is a classification method that makes a set of hierarchical decisions on the feature values formed in a tree-like structure. Any decision splits the tree based on a criterion in a way that the training data is divided into two or more branches. The goal is to find the best split criterion by which the number of mixing the class variables in each branch of the tree is reduced as much as possible [33]. There are three classical algorithms for decision tree including ID3, C4.5 and CART (Classification and Regression Trees). These algorithms use two splitting criteria called as 'Entropy' and 'Gini'. Among three classical algorithms for decision tree which are known as C4.5, ID3 and CART we have applied CART algorithm in order to build our data model. CART has advantages over the other algorithms in terms of reducing over-fitting and the ability of handling incomplete data [34]. It also builds models for regression as well as classification. CART uses Gini criterion for splitting. An optimized version of CART that has been implemented by scikit-learn is used in this work.

2- Naïve Bayes

Naïve Bayes classifier is the simplest form of a Bayesian Network. It is termed 'naïve' because it assumes that all attributes are conditionally independent. In spite of this controversial assumption which is used to simplify the process of modelling, Naïve Bayes is a fast classifier and has a great performance in practice for many domains [35]. We have applied Gaussian Naïve Bayes classifier implemented in the scikit-learn library therefore, the probability of the features is assumed to be Gaussian.

3- Logistic Regression

Logistic regression is an analytic method for classification problems. It is able to model scenarios with two or more

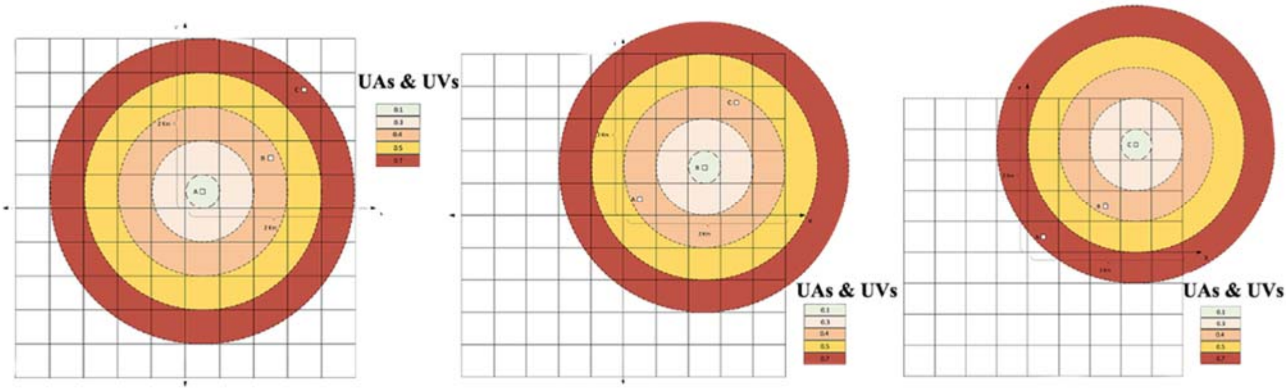


Figure (3): Uncertainty Areas for three defined PoIs, from PoI_1 to PoI_3 from left to right.

possible discrete outcomes. It uses a probabilistic classifier and maps the feature variables to a class-membership probability. The most common form of logistic regression builds data-driven models with binary outcomes (e.g. Access/Deny). In this work we have used a logistic regression classifier with binary outcomes.

4- Support Vector Machine (SVM)

The support vector machine (SVM) is one of the most robust and widely used binary classification algorithms. The goal of the SVM optimization program is to determine the separating hyperplane which maximizes the distance between the closest training samples to it (the *support vectors*) [36]. This reduces the misclassification error whilst maximizing the generalization capability for test datasets. In addition, when the training set is non-linearly separable as it is the case in this study, SVM is combined with the kernel trick to expand the space implicitly, facilitating the linear separability for the two classes [36]. In this research we applied the support-vector classification algorithm from scikit-learn library in order to build our access prediction model.

C. Validation

Cross-Validation has been used to validate the model optimized by each of the above machine learning methods. Cross validation is the widely used approach to evaluate the generalizability of proposed models [37]. In order to conduct the process of cross-validation, 10-fold cross validation was chosen and 10% of dataset was assigned to the test split. We set the shuffling data feature to true in order to increase the chance to find the best fit model parameters and improve the generalizability of the generated model.

V. Results and Discussion

Formally each sample request in our dataset is mapped to one element of the set {Deny, Access}. Based on this notation the performance of the applied classification algorithms is evaluated in terms of the following metrics: (i) Accuracy (ii) *Precision* (iii) *Recall* and (iv) F1.

Table (4) shows the performance results of these four prediction models in details. As shown, SVM and logistic regression have the same results in terms of accuracy, precision, recall and F1. This was expected due to the method of optimization that these two algorithms are using. In other words, the way of updating the model parameters in logistic regression is the same as the way of updating the weights in neural network model. These two algorithms learned from their mistakes in classification in order to update.

Table (4): Performance results for prediction models

	Accuracy	Precision		Recall		F1	
Decision Tree	88.02% (*SD=0.91%)	0	1.00	0	0.29	0	0.45
		1	0.87	1	1.00	1	0.93
Naïve Bayes	80.94% (*SD=1.17%)	0	0.44	0	0.37	0	0.43
		1	0.87	1	0.90	1	0.89
Logistic Regression	88.14% (*SD=0.86%)	0	1.00	0	0.29	0	0.45
		1	0.87	1	1.00	1	0.93
SVM	88.14% (*SD=0.86%)	0	1.00	0	0.29	0	0.45
		1	0.87	1	1.00	1	0.93

**SD=Standard Devaition*

We have also applied a sensitivity analysis using Receiver Operator Characteristic (ROC) curve [38]. As shown in Figure (4), ROC curves for logistic regression and SVM dominate the other curves. Naïve Bayes model shows lower performance with respect to true positive and false positive rates than the other prediction models.

Comparing the results derived from top three classification methods in terms of accuracy indicates that logistic regression and SVM show the better performance in terms of model accuracy in comparison with decision tree. On the other hand, computational complexity of logistic regression algorithm is lower than decision tree and SVMs [39] therefore, using prediction model created by logistic regression algorithm is recommended in any scalable and dynamic environment like fog and mobile edge computing in the context of IoT due to the limitations of devises in those environments in terms of processing capability and energy consumption.

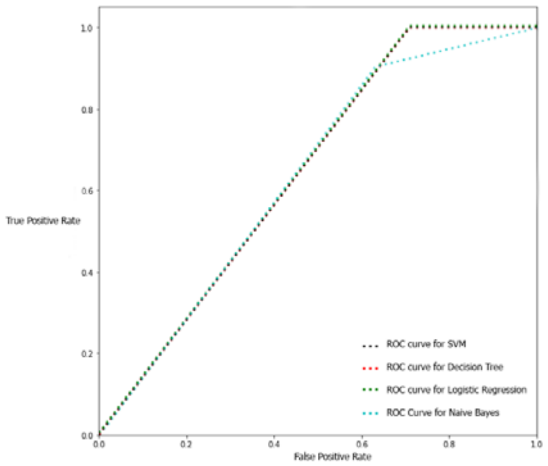


Figure 4: Aggregated ROC curves for Class: Access

VI. Conclusion

Traditional and emerging access control models are not applicable to scalable, dynamic and heterogenous computing and communication paradigms like fog and mobile edge computing. On the other hand, resilient access control approaches like BTG and optimistic access control cannot guarantee the security if the number of policy overriding goes beyond the threshold. Moreover, traditional risk aware access control approaches do not fit into scalable and heterogenous environment like fog computing in the IoT era.

Despite the fact that several approaches have been proposed to address scalability, dynamism and heterogeneity of access control in the IoT era, uncertainty in authentication remains as a neglected challenge. For this reason, in this research we have defined and modeled uncertainty in authentication. In doing so, we have proposed a novel data-driven model as the extension of ABAC. Our model is able to handle mobile users/devices for fog and mobile edge computing environment. The model was built using four robust classification algorithms namely, logistic regression, SVM, Naïve Bayes and decision tree. In order to train and test our model we have developed a dataset based on the findings of the state-of-the-art researches. The results showed that prediction model created by logistics regression has the better performance in terms of accuracy (88.14%) and has the lower computational complexity than the other algorithms.

The future step of this work is to consider the other elements of indeterminacy like ambiguity besides uncertainty. We will also apply other classification algorithms like ANN, Random Forest and K-Nearest Neighbors to make comparison among classification algorithms. Moreover, implementing the proposed model in a real-world scenario will be our priority for the future work.

References

[1] Shanhe Yi, Zhengrui Qin, and Qun Li, "Security and Privacy Issues of Fog Computing: A Survey," in *International Conference on Wireless Algorithms, Systems, and Applications, Springer* , 2015.

[2] MITHUN MUKHERJEE, RAKESH MATAM, LEI SHU et al., "Security and Privacy in Fog Computing: Challenges," *IEEE Access,* vol. 5, pp. 19293-19304, 2017.

[3] Stojmenovic, I., Wen, S., "The fog computing paradigm: Scenarios and security issues," in *IEEE Federated Conference on Computer Science and Information Systems* , 2014.

[4] Rodrigo Romana, Javier Lopez, Masahiro Mambo, "Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems,* vol. 78, pp. 680-698, 2018.

[5] William Stallings, "Access Control," in *Computer Security, principles and practice*, Pearson, 2017.

[6] Aafaf Ouaddah, Hajar Mousannif, Anas Abou, Elkalama Abdellah , "Access control in the Internet of Things: Big challenges and new opportunities," *Computer Networks,* vol. 112, pp. 237-262, 2017.

[7] M. H. Ibrahim, "Octopus: An Edge-Fog Mutual Authentication Scheme," *International Journal of Network Security,* vol. 18, 2016.

[8] P. Hu, H. Ning, T. Qiu, H. Song, Y. Wang, and X. Yao, *IEEEE Internet of Things,* 2018.

[9] D.R. dos Santos, C.M. Westphall, C.B. Westphall, "Risk-based dynamic access control for a highly scalable cloud federation," in *IEEE Proceedings of the Seventh International Conference on Emerging Security Information, Systems and Technologies*, 2013.

[10] D.R. dos Santos, C.M. Westphall, C.B. Westphall, "A dynamic risk-based access control architecture for cloud computing," in *IEEE Network Operations and Management Symposium (NOMS)*, 2014.

[11] Daniel Ricardo dos Santos, Roberto Marinho,Gustavo Roecker Schmitt, "A framework and risk assessment approaches for risk-based access control in the cloud," *Elsevier Journal of Network and Computer Applications,* vol. 74, 2016.

[12] S. Savinov, "A Dynamic Risk-Based Access Control Approach: Model and Implementation," *PhD Thesis, University of Waterloo,* 2017.

[13] F. Salim, "Approaches to Access Control Under Uncertainty," *PhD Thesis, Queensland University of Technology,* 2012.

[14] A. Ferreira, R. Cruz-Correia and L. Antunes, "How to Break Access Control in a Controlled Manner," in *19th IEEE International Symposium on Computer-Based Medical Systems*, 2006.

[15] Htoo Aung Maw, Hannan Xiao, Bruce Christianson, and James A. Malcolm, "BTG-AC: Break-the-Glass Access Control Model for Medical Data in Wireless Sensor Networks," *IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, ,* vol. 20, no. 3, pp. 763-774, 2016.

[16] Schefer-Wenzl, S., & Strembeck, M., "Generic Support for RBAC Break-Glass Policies in Process-Aware Information Systems," in *28Th Annual ACM Symposium on Applied Computing*, 2013.

[17] D. Povey, "Optimistic Security: A New Access Control Paradigm," in *ACM workshop on New security paradigms*, 1999.

[18] Molloy, I., Dickens, L., Morisset, C., Cheng, P. C., Lobo, J., & Russo, A., "Risk-Based Security Decisions under Uncertainty," in *Proceedings of the Second ACM Conference on Data and Application Security and Privacy*, 2012.

[19] Jason R.C. Nurse, Sadie Creese, David De Roure, "Security Risk Assessment in Internet of Things Systems," *IT Professional ,* vol. 19, no. 5, pp. 20-26, 2017.

[20] Wei Zhou, Yan Jia, Anni Peng, Yuqing Zhang, and Peng Liu, "The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, and Challenges Yet to Be Solved," *IEEE Internet of Things Journal,* pp. 1-11, 2018.

[21] ELISA BERTINO, KIM-KWANG RAYMOND CHOO, DIMITRIOS GEORGAKOPOLOUS, SURYA NEPAL, "Internet of Things (IoT): Smart and Secure Service Delivery," *ACM Transactions on Internet Technology,,* vol. 16, no. 4, pp. 22-29, 2016.

[22] Francesco Restuccia, Salvatore D'Oro and Tommaso Melodia, "Securing the Internet of Things in the Age of Machine Learning and Software-defined Networking," *IEEE Internet of Things,* vol. 1, no. 1, p. IEEE Early Access Service, 2018.

[23] H. Reza Ghorbani ; M. Hossein Ahmadzadegan, "Security challenges in internet of things: survey," in *IEEE Conference on Wireless Sensors (ICWiSe)*, 2017.

[24] Mario FRUSTACI ; Pasquale PACE ; Gianluca ALOI ; Giancarlo FORTINO, "Evaluating critical security issues of the IoT world: Present and Future challenges," *IEEE Internet of Things Journal* , pp. 2327-4662, 2017.

[25] P. D. Gallagher, "NISP SP800-30 Guide for Conducting Risk Assesment," NIST, 2012.

[26] T. Moses, " “Extensible Access Control Markup Language (XACML)," OASIS, 2013.

[27] "User-Computer Authentication Associations in Time," Los Alamos National Laboratory, [Online]. Available: https://csr.lanl.gov/data/auth/. [Accessed 13 02 2019].

[28] V. Lohweg, "banknote authentication Data Set," Center for Machine Learning and Intelligent Systems, University of California, [Online]. Available: https://archive.ics.uci.edu/ml/datasets/banknote+authentication. [Accessed 13 02 2019].

[29] Uthpala Subodhani, Premarathne, Ibrahim Khalil, Mohammed Atiquzzaman, "Location-dependent disclosure risk based decision support framework for persistent authentication in pervasive computing applications," *Computer Networks,* vol. 88, pp. 161-177, 2015.

[30] GANESH CHANDRASEKARAN, NING WANG, MASOUD HASSANPOUR, MINGWEI XU AND RAHIM TAFAZOLLI,, "Mobility as a Service (MaaS): A D2D-Based Information Centric Network Architecture for Edge-Controlled Content Distribution," *IEEE Access,* vol. 6, 2018.

[31] Frans Ekman, Ari Keranen, Jouni Karvo, Jörg Ott, "Working day movement model," in *ACM Proceedings of the 1st ACM SIGMOBILE workshop on Mobility models*, 2008.

[32] Ari Keränen, Jörg Ott, Teemu Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *ACM Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, 2009.

[33] C. C. Aggarwal, Data Mining, The Text Book, Springer, 2015.

[34] M Balamurugan ; S Kannan, "Performance analysis of cart and C5.0 using sampling techniques," in *IEEE International Conference on Advances in Computer Applications (ICACA)*, 2016.

[35] Sebastian Raschka, Vahid Mirjalili, Python Machine Learning, Machine Learning and deep learning with python, scikit-learn and TensorFlow, Packt, 2017.

[36] Bernhard Schölkopf and Alexander J. Smola, Learning with Kernels, Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, 2002.

[37] Ian H. Witten, hor), Eibe Frank, Mark A. Hall, Christopher J. Pal , Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, 2016.

[38] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters,* vol. 27, no. 8, 2006.

[39] Anna L. Buczak, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys & Tutorials* , vol. 18, no. 2, 2016.

[40] W. Stallings, "Access Control," in *Computer Security, principles and practice*, Pearson, 2017.