Access Control Based on Log File for Internet of Things Devices

Arwa Aloqbi, Manal Alshammari, Amal Alatawi, Amer Aljaedi, Adel R. Alharbi

Abstract**: *The use of Internet of Things devices has lately increased significantly, leading to the management of a diverse set of nodes and a vast number of data. Most Internet of Things nodes have limited resources and are vulnerable to a variety of threats and failures. Therefore, numerous novel techniques have been conducted to secure resource-constrained devices such as access control. In this work, we proposed an access control mechanism by using the user log files when they interact with their Internet of Things devices. Where it is possible to define and enforce access control restrictions and follow logs through log files to monitor the user accessing behaviors. This mechanism can be applied as an extra security layer along with any traditional user authentication access control to have the effective and accurate access control to prevent intrusion reveal information in the Internet of Things devices. To do this, we developed three Internet of Things applications on mobile, table, and website pages with different functionalities and goals to store the user log file features. We collected a large-scale date-set from over a thousand participants. Three machine learning algorithms: J48, Part, and Naive Bayes are applied and compared to predict the legitimate users. Several experiments were performed with significant results.***

Keywords**: *Internet of Things; Access Control; Attribute; Logs File; Machine Learning.***

I. INTRODUCTION

Recently, Recently, the use of Internet of Things (IoT) devices has increased dramatically, which has led to dealing with a variety of devices and a huge amount of data as usage has increased, it has also raised with it a set of rather challenging problems for the correct use of materials in IoT devices. These devices are connected to each other in smart environments by using low-power wireless communication, servers, and cloud computing, etc. Most of the **IoT** devices have limited resources and are vulnerable to different types of threats and failures. Therefore, access control was one of the most important factors we have in the development of IoT, traditional access control models may not be suitable for access control in an open network environment like IoT, because access control process cannot be implemented smoothly due to the diversity of their access policies. New security challenges have been addressed and need cutting-edge solutions to be implemented to make the data protected from being tempered or manipulated by attackers. Data privacy must be properly handled to avoid the sensitive data of being misused by attackers in a variety of environments [1]. Numerous novel techniques have been conducted to secure such resource-constrained devices. One of the most technique used is access control to validate if user access is legal to access some services. Access control is a mere a set of rules that is used to control the access to other nodes and information. Different type of access control includes attribute-based access control (ABAC), role-based access control (RBAC), and fine-grained access control (FGAC). Moreover, many of the state-of-the-art studies have used ABAC method to authenticate a user based on the attributes of the devices, users, and environments. The ABAC is a promising technique because of its ability to provide flexible and dynamic access policies [2]. A user grants access to information and resources when the user's attributes satisfy the prepared policy specifications. The authors in [3] used NIST (National Institute of Standards and Technology) Next Generation Access Control (NGAC) to control access in the IoT environments. The proposed model can learn the user's attributes and then be able to detect access anomalies in smart environments. In this work, a user device receives periodic wireless beacons from static nodes, then the user's attributes will be recorded by a mining service to learn the access control rules and be able to detect anomaly access. Furthermore, the ABE method is a technique that is used for data access control for user privacy protection [4], is public key encryption mechanism, and it supports encryption and decryption procedure that is more flexible than traditional cryptography. ABE is public key encryption mechanism. On the other hand, several machine learning and deep learning techniques have used to validate the access of resources and they can be classified into two groups: legal or illegal access. Such techniques include CNN, decision tree, K-Nearest neighbor, logistic regression, random forest, naive bays [5], and many other which are considered as the important methods to check the validity of logs in the proposed system of this paper. Despite the significant advances that machine learning techniques have made in identifying abnormal, there are still several issues that limit protecting user privacy, including a lack of collaboration between detection algorithms and other security mechanisms such as access control [6]. Because of the isolation, the detection models are unable to leverage accessible user characteristics to rule out false alarms.

Manuscript received on 08 June 2022 | Revised Manuscript received on 14 June 2022 | Manuscript Accepted on 15 July 2022 | Manuscript published on 30 July 2022.

* Correspondence Author

Arwa Aloqbi, College of Computing Information Technology University of Tabuk, Tabuk 71491, Saudi Arabia: 421009294@ut.edu.sa

Manal Alshammari, College of Computing Information Technology University of Tabuk, Tabuk 71491, Saudi Arabia: 421009296@ut.edu.sa

Amal Alatawi, College of Computing Information Technology University of Tabuk, Tabuk 71491, Saudi Arabia: 421009234@ut.edu.sa

Amer Aljaedi, College of Computing Information Technology University of Tabuk, Tabuk 71491, Saudi Arabia: aaljaedi@ut.edu.sa

Adel R. Alharbi*, College of Computing Information Technology University of Tabuk, Tabuk 71491, Saudi Arabia: aalharbi@ut.edu.sa

Retrieval Number: 100.1/ijrte.B70940711222
DOI: 10.35940/ijrte.B7094.0711222
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

- Our work has many contributions for access control based on log files with machine learning space, we can summarize it as below:
- Develop three IoT software apps, which are: website app, mobile app, and table app to collect user's log files. Each of these apps have different purposes and functionalities.
- Collect a large-scale log file dataset using these apps from users living into different countries and cities around the globe to get a valid access control data.
- Create a machine learning schema from extracting log file features and filtering them and apply a set of popular machine learning algorithms.

II. LITERATURE REVIEW

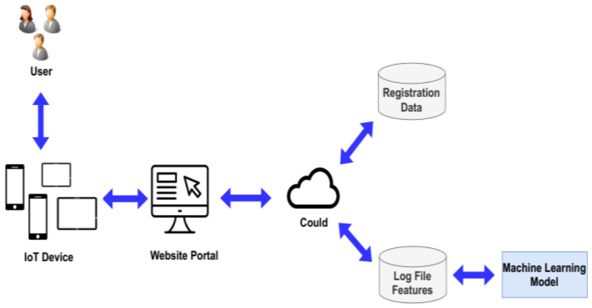
In this chapter, we will present some concepts concerning The field of user rights-based policies for mobile applications relates to the work done in the same areas of access control policies and technologies for access. The IoT is a system that connects computing devices without the aid of human or computer interaction. Khan, Md Abbas et al. [7] proposed a system that provides security, an application-based remote access control door lock (RACDLS) and a short-range wireless communication called Near Field Communication (NFC). RACDLS creates a two-sided authentication system instead of a one-sided authentication like traditional systems. In a traditional system, to maintain both data integrity and confidentiality, the encryption technique we consider a 512-bit arithmetic hash function. In contrast, implement AES-192 to encrypt hashed data. In addition, machine learning shows employee activity performance and predicts the accuracy of the model. The objective of this paper is to ensure the security of remote access control as well as allow reporting by both parties, ease of use and accessibility. Liu, Aodi and et al. [8] proposed a scheme for an efficient decision-making engine based entirely on machine learning (EPDE-ML). The proposed scheme converts an attribute-based access control request into a vector of permission decision, turning from an access control permission decision problem to a binary classification problem that allows or denies access. Use the random forest algorithm to help build a vector decision classifier to create an efficient permission decision engine. The experimental results showed that the proposed method can achieve permission decision accuracy of about 92.6\% on the test data set, and the permission decision efficiency is much higher than the standard method. The improvement in their performance becomes more evident as the scale of the policy increases. There is a very extensive literature Shebaro and et al. [9] introduce context-based access control (CBAC) for Android systems. A mechanism in their work [9] that allows smartphone users to set configuration policies for application use. Through the CBAC mechanism, users can, based on the restriction of privileges in specific places such as work, regain privileges when using the device. Elsewhere, this change in device privileges is automatically applied once a user's device matches a predefined context for a user-defined policy. The user can also define a default set of policies to be applied when the user is in an unspecified location. At the time of installation, configured policy restrictions are determined by the accessible device resources, services, and permissions granted. These policies define the services provided by the device and limit access to device and user information. Policy restrictions are context related and configured by the device user. We determine the context according to the place and time by determining the location primarily through visual Wi-Fi access points (APs) and their respective signal strength values that allow us to distinguish between areas, as well as GPS and cellular triangulation coordinates if available, implements CBAC policies on Android We include a tool that allows users to locate physical locations such as home or work using captured Wi-Fi parameters. Once the user defines device policies that define device and application privileges according to context, the policies will be applied automatically when the user is within a predefined physical location and time interval. Suggested method and permission system Android's permission system controls which app has the privilege to access certain device resources and data.

Application developers specify the permissions they need in the AndroidManifest.xml file Each app announces the permissions listed in its AndroidManifest.xml file at the time of installation and asks users to either grant all required permissions to continue with the installation or cancel the installation. Android's permission system only allows users to grant or deny some required permissions, which limits user control over access to the app. Idrissi, Hind, and et al. [10] in their work propose a scheme that tracks a new policy for controlling access using mobile agent technology. the mobile agent transfers all its resources (code, data, and execution state) in order to perform independently without going back to the original platform. To meet the security requirements at all levels (network, system, application) of the information system. The proposed scheme takes advantage of the mobility aspect of agents to avoid the flaws of classic client/server connections and uses cryptographic mechanisms such as encryption and the nature of a digital signature to ensure authentication, identification, confidentiality, and integrity. Moreover, Chang and Victor [11] provided a developed a framework known as the Cloud Computing Certification Framework (CCAF) which is dedicated to securing cloud data. This paper explains the overview, rationale, and components of CCAF for data security protection. Hou, Yichen, and et al. [12] proposed a Data Security Enhanced Micro- Access Control (FGAC) mechanism to ensure data security is ensured while accessing data in mobile edge computing. At the FGAC, a precise dynamic scheme for grouping trusted users was first designed based on trait and clause theory. Secondly The scheme is combined with the traditional role- based access control mechanism to assign roles to users based on user group trustworthiness. Then, based on the attribute matching, the user authentication further checks whether the user is allowed to make accesses to achieve accurate data protection. Experimental results show that FGAC can effectively identify malicious users and perform mass modifications, while achieving accurate access control and ensuring data security during the data access process in mobile computing. In FGAC, all users are divided into different groups, and each user accesses data resources according to the role assigned by the user group’s credibility.

We consider collusion attacks and self-improvement attacks initiated by internal attackers. Attackers can increase their access to important resources through collaboration, thereby threatening data security. The specific attack is defined as: Collusion Attack is where multiple attackers can cooperate and provide false information to increase the reputation value of malicious users and reduce the reputation value of normal users, thereby affecting the security level of users. Self-promotion Attack is where attackers try to increase their reputation by mistake by providing false information or exploiting calculation loopholes, thereby improving their security level. Liu and et al. [13] propose a systematic framework for dealing with security and privacy requirements. The framework supports a range of analysis techniques. In particular, the analysis of attacks helps identify abusers. Countermeasures analysis supports the dynamic decision- making process of defensive system players in addressing weaknesses and threats. Finally, access control analysis bridges the gap between security requirement models and security application models. The framework is illustrated with an example that includes security and privacy concerns in the design of agent-based health information.

III. METHODOLOGY

With the acceleration of technologies and the almost complete reliance on IoT devices, the need for integration between applications and computing services has emerged. From here, we find it necessary to restrict applications with some privileges to different users according to their entitlement and their daily tasks to ensure the security of sensitive information for users from intrusion. The fundamental concept behind our system is to employ attribute access control on a website portal-based machine learning algorithm to distinguish between regular and anomalous logins. The suggested system was divided into two phases: registration and log data collection for a while and then a new data set is created to be used for future in a machine learning model until it recognizes users' logs successfully and be able to do prediction phase. As it is shown in Figure 1, the initial step, users must register with the system via a website portal that could be in the cloud, which records certain IoT device log file features such as: timestamp (e.g. date and time), city, country, browser type, and etc. These features extracted and stored into a database for every user using our IoT developed applications. Finally, the collected IoT data-set log features will be used into the machine learning model to distinguish legal and illegal user as an extra security layer along with the user username/password that already stored in the website portal database system. The machine learning model will role as a prediction phase to the IoT devices users.



Methodology overview (Figure 1)

A. Data Collection

We built our study based on three IoT developed applications. One app was on mobile devices. Second app was on tablet devices. The last application was a website page which can run in any IoT devices. Next, we explain the goals and functionalities of these applications.

1) **Mobile application**: A program developed in the Android environment [14], the fitness app is designed to serve users who wish to continue exercising and choose trainers to. It is an application with a natural communicate with them interface, familiar to the user, and is characterized by ease of login. The first use of the application requires a new user registration. The application was published on Google Play on November 8, 2021, and it is now possible to download and install it for the Android operating system. We were able to collect 100 users.

2) **Tablet application:** A new Android application [15] for employment has been developed to start a connection between a job seeker and a service seeker, this application aims to provide a way to explore and add services to the service applicant and a job description for the job seeker has been implemented, through this application the job seeker and services can communicate through social media channels such as WhatsApp, in addition to the possibility of direct contact for both. Several authentication procedures have been implemented to ensure the reliability and trust of the user, so each user must create a new account. Through the registration steps, we have collected information about the user along with GPS information such as location and IP address where this information is stored in the database, as we considered the GPS as information Important, the app was published on November 7, 2021, in Google Play and has collected 1,000 users

3) **Website application**: We developed a website page that consisted of registration and collecting logs data. The main idea behind this system is to use website portal-based access control to discover. Users are required to register in the system using the website portal, which stores some user data in the log file such as: name, city, country, the browser type, operating system, etc. The data is used to identify users when they log into the system. For example, When the user logs into the system, a new record will be stored in the log file cloud database. Many users have registered with this system and logged in multiple times from different IoT devices. We asked random users to participant with our website started in November 15, 2021 and we stored their data-sets for our further studies.

B. Feature Extraction

We can divide the features that were obtained in our work in two categories:

1) **User experience features inside the app**: These features are different than user feature behaviors (e.g., keystrokes), they are about the experience with the app itself. It concentrates on features where user interact with app tasks. For example, the most task that user was used or clicked during the using of the app experience.

2) **User experience features outside the app**: These features are captured based on the data source outside the app like the GPS. This allows us to obtain some features like the user county, city, and IP address. These features are requisite in any log file systems.

Table-I: Feature extraction summary

Data #	Application	# of instances	# of features
D1	Mobile	529	52
D2	Tablet	5,040	33
D3	Website	200	20

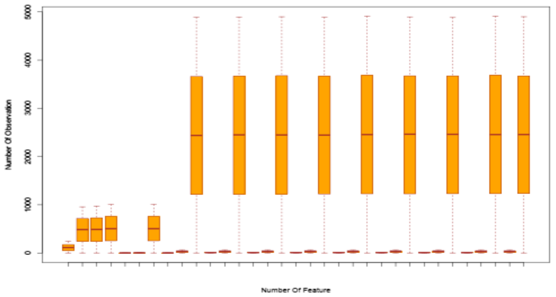
Table-II: Description of the most extracted features

#	Name	Description
1	Country	Indicates the country of the user at the time of loggingin
2	City	Indicates the city of the user at the time of logging in
3	Time-stamp	Indicates the time of the user device in the log in process
4	IP address	Indicates the IP address of the user at the time of logging in
5	First action	Indicates the first action from the user
6	Most used	Indicates the most action from the user
7	Admin	Indicates to adding the user to admin
8	Coach	Indicates to adding the user to coach
9	Exercise	Indicates to adding the user to Exercise
1 0	Logout	Indicates to logout the user from application

C. Statistical and Data Analyzes

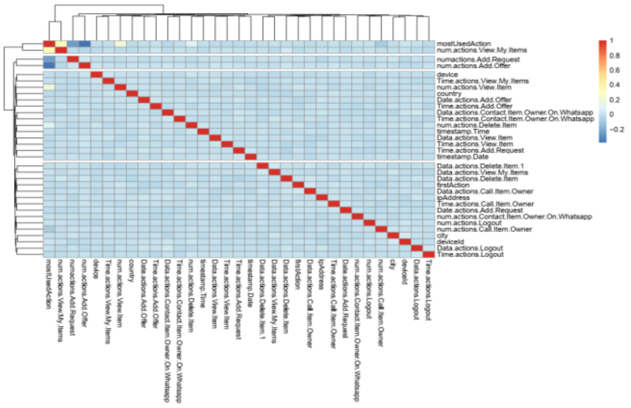
In here, we provide some of the statistical analyzes on the log file features that were captured throw the date collection process.

1) **Box Plot:** A box chart is one of the methods used to visualize data collected in an interpretive data analysis. Visually shows distribution of numerical data and deviation by displaying quartile data (or percentages) and averages [15]. The median box plot uses approximate quartiles and lower and higher data points to convey the level of distribution, spread, and uniformity of data values. They can also be easily revised to identify external data values and can be easily generated manually. Figure 2 presents the box plot above, where the first feature shows that the first quartile (Q1) is 1 and the third quartile (Q3) is 600, and the lower bound (Q0) for this data set is 0. The median (Q2) for this data set is 10 From the second feature to the fourth feature also, the first seventh quartile (Q1) is 200 and the third quartile (Q3) is 600, the minimum (Q0) for this data set is 0, the median (Q2) for this data set is 300. In general, there are no outliers, and the data follow a normal distribution. It also shows us that the identical median [16].



Feature Box Plot (Figure 2)

2) **Correlation Matrix**: This part describes the relationship between features by using its correlation. Correlation generally is used to evaluate the association between two or more variables. The thermal map of the link is a way to graph the link matrix, which in turn represents the link between our different variables [17]. In figure 3, shows the relationship between features by using its correlation matrix is a statistical technique by which we get an idea about the data set, so that we can analyze the data set and make a decision according to it. As we can build a machine learning model according to the result, so this technique is very useful. This is the most common statistical method as well. Correlation values range from 1 to -1: If the value is >0, the correlation is positive. If the value is <0, the correlation is negative. A value of 0 means that the correlation is independent. The figure also shows the correlation matrix for our data set. Main Diameter elements have correlation values of 1 and are colored red "Self-correlate column". The rest of the values range in the direction of blue color gradients based on the correlation values that are close to independence. Thus, we need all the attributes of the dataset [18].



Feature Correlation Matrix (Figure 3)

3) **Data Variances Analysis:** The method was used to produce the vector-wise z-score of the data used as input, with a center of zero and a standard deviation of one. Some other experiments have been conducted by using three columns every time. Besides, one experiment has been done to test the analysis of variance (ANOVA) for all columns. We noticed that the value of *p*-value is very small and equal to 0.0012. These very small *p*-values indicated that there is a high significant difference between the sample means. Therefore, we reject the null hypothesis that there is no difference in means across the proposed data classes [19].

D. Supervised Classification

In this project, we applied three different classification famous algorithms.

1) **J48:** J48 algorithm is defined as the optimization of C4.5. The J48 algorithm is considered one of the best machine learning algorithms, which is why it checks data categorically and continuously [20].

For example, it takes up more space in memory and depletes performance and accuracy in classifying medical data. It used for Generating a Trimmed or Uncompressed Decision Tree C4.5 in a Waikato Knowledge Analysis (WEKA) Work Environment Developed by Ross Quinlan 1993 [21]. It is an open-source Java implementation of simple C4.5 decision tree algorithms. J48 is an extension of ID3 with additional features are calculating missing values, pruning decision trees, continuous attribute value ranges, derivation of rules, etc.Usually, we construct a decision tree from the data set using information acquisition and examine the same results, after selecting a feature to segment the data. The algorithm then, iterates the same previous work on smaller subsets. The algorithm stops performing partitioning if all instances in a subset belong to the same class. Then the final step is a leaf node is created in a decision tree to choose that class [22]. The output from the J48 algorithm is a decision tree. This tree is defined as the same tree as the tree structure that consists of different nodes, for example the root node, intermediate node, and the leaf node. Each node in the tree contains a decision and ultimately the decision leads to the result [23].

2) **Partial Decision Tree (PART) rules:** This algorithm shows in the first place as a child of the Decision Tree (DT) [24]. According to Berger, and et al. [25] proposed a approach to subset of the feature of the all feature can represent the best practice to achieve a remarkable results while reserving a best accuracy. This proposed model came to measure the co-related sub-set feature, which can be minimize and the rule sets stay achievable for metrics measurable. Where they focused on the ability to verified two different document representation and four different text classification algorithms. Frank, and et al. [26] claimed that individual rules work separately and repeatedly can achieved without any required for the global optimization. In other words, they are inferring rules one rule at time, by avoiding the need for a global optimization conducting by DT algorithm. In fact, the main different in their approach and the traditional DT approach, that their approach combining two major paradigms for generating rule (creating rules) using the traditional DT algorithm. Then, separate and conquer the learning rules. By implementing, each rules separately. The way that the Berger, and et al. approach works is to separate the algorithm into two main parties the first part focus on identifies the whole feature related to the data where in their research a huge number of features extracted from the text corpora. Therefore, the number of features enforce them to use one of the three popular approaches in feature selection. They choose filter approach, which also depend on the DT learner using PART. Rule learners are prominent representatives of supervised machine learning approaches. This type of learner tries to induce a set of rules for a collection of training instances. These rules then applied on the test instances for classification purposes. They combine C4.5 and RIPPER, which are two well-known learner. First, an initial rule set is determined, and second, these rules adjusted or discarded according to a global optimization strategy. PART has feature selection method that decreases the number of features that have already been lowered using traditional feature selection criteria [24]. PART also develop a set of rules developed during the training phase of PART. Each rule has an arbitrary number of characteristics that linked together using Boolean operators and a class. In this example, binary feature weighting employed, thus each mentioned attribute is either explicitly present or missing in each instance. Following the completion of PART's training phase, the algorithm walks through each rule in the Ruleset and extracts all features included in the rule. The union of the newly extracted features (Features) and the reduced feature set (RedF) is then determined. Finally, the whole set of reduced characteristics is calculated [27].

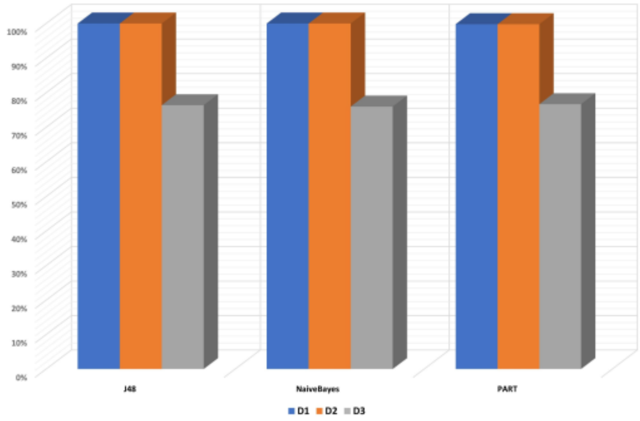
3) **Naive Bayes:** Naive Bayes classifiers are a collection of classification algorithms based on Bayes' theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e., every pair of features being classified is independent of each other [28]. The Naive Bayes models are one of the most basic and yet consistently effective classifiers. The assumption that all characteristics used to characterize an instance are conditionally independent given the class of that instance is, however, a weakness in these classifiers. When this assumption is violated (as is frequently the case in fact), "information double-counting" and interaction omission can degrade categorization accuracy. From the historical point of view, we can conclude that Naïve Bayes Classifier history consider the main classification algorithm discussed and used over decades. This way of work according to the power and efficiency of handling any classification problem because of the shuffle of the consideration of co-related of attributes [29]. In Gaussian Naive Bayes, continuous values associated with each feature are assumed to be distributed according to a Gaussian distribution. A Gaussian distribution is also called Normal distribution. When plotted, it gives a bell-shaped curve, which is symmetric about the mean of the feature values. Support the assumption of algorithms enabled to classify and learn 'such as q is a random variable that points to a classifier like V that is vector of random variables representing a specific classification. While the represent a specific direction for the observed value. Make a value test according to the probability of each classification [30].

IV. EXPERIMENTAL RESULTS

In this section, we attempt to apply different data validation techniques. We used Weka the machine learning software tool [31]. In all the experiments, we applied the same defaults seeding and tuning settings for the classification models to avoid any bias in the results. Several classification evaluation metrics were implemented such as accuracy, ***F***-Measure, ROC Area, etc. as explained in detail in [32].

A. Classification model comparison

In this experiment, we used the ***10***-fold cross validation method for all the three classifiers. ***k***-fold cross validation is a method for evaluating predictive models based on machine learning, by dividing the original data set into a training set for training the model and a test set for its evaluation. This technique randomly divides the original data set into *k* subsets of equal size. The first subset is taken as the test set of the model, and the remaining subsets (*k*-1) are used as the training data set for the model. The data set will be divided into ten subgroups of equal size. This technique should be applied ten times in a row wherein the first subset in the first round should be considered as a test set (test fold) and the remaining subsets (*k*-1 subsets) as training sets (training folds). We test the model and record the value we get. In the second round, the second subgroup will be considered as the test group and the remaining nine groups as training groups, and the model accuracy result will be recorded in this round as well. After the cross-validation process is over, the accuracy of the model can be obtained by averaging the *k* -fold results [33]. As we can see in figure 4 the accuracy results of our three compared classifiers, where the J48 classifier outperformed the other classifiers for all the collected data sets. There are smellier performance results between the J48 and Part classifies that because the mechanism of operation on tree architecture, thus the J48 classifier is much faster, but it may cause an over-fitting issue for larger data sets.

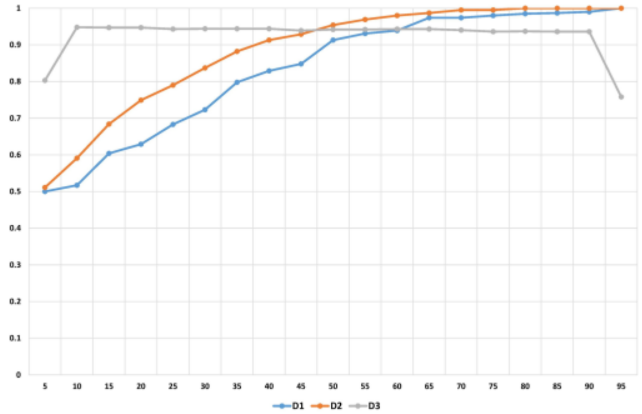


Classifiers accuracy comparisons (Figure 4)

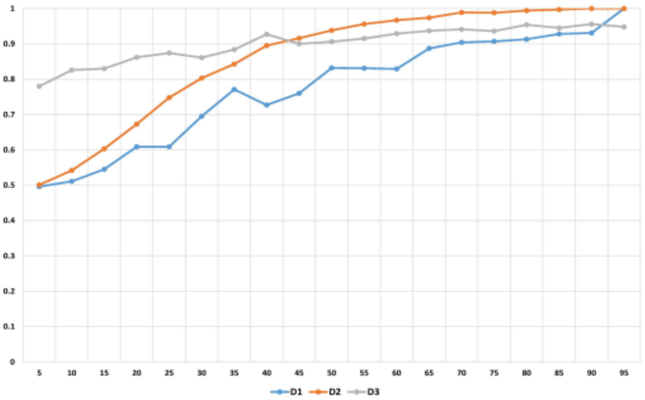
B. Different training sets validation

This experiment applied various split of the training sets to fed the classifier and remaining sets used as testing or validating. We have stared up with small portion of training data set, then increasing it by 5% each time in order to valid the classification models in different situations. We attempt to find the perfect training split of our collected data sets. We obtain our results using the Receiver Operating Characteristic (ROC) area, which is a method widely used to measure the ability of any classifier to distinguish between classes [34].

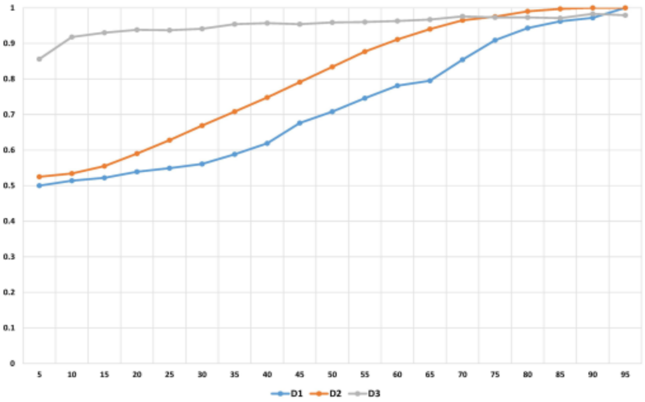
Figure 5 presents the j48 classifier ROC area results from splitting the data into training small portions increment it each time with 5% until 95% to observer who the classifier will react with smaller amount of the training data. As we can notice that the classier stats with performance of 0.5 and increment as the training split increases until reached to the 55%, after the ROC results becomes stables as we add more training data into it. In Figure 6 and 7 same evaluation concept applies of splitting the data and measure the other two classifiers are reacting. Figure 6 presents the Part classifier ROC area results. As we cab notice that the classifier stats with performance of 0.5 and increment as the training split increases until reached to the 75%, after the ROC results becomes stables as we add more training data into it. Figure 7 presents the Naive Bayes classifier ROC area results. As we cab notice that the classifier stats with performance of 0.5 and increment as the training split increases until reached to the 80%, after the ROC results becomes stables as we add more training data into it.



J48 classifier ROC training splatting results (Figure 5)



Part classifier ROC training splatting results (Figure 6)



Naive Bayes classifier ROC training splatting results (Figure 7)

V. CONCLUSION AND FUTURE WORK

To work with the surrounding areas, sophisticated software and hardware are required. IoT devices are becoming more popular, yet they are prone to failure when connecting to unfamiliar surroundings. Access control is a strategy that requires enforcing a set of regulations that regulate how people get access to resources and information. The suggested model in this study is primarily capable of extracting log information when users use their tablets to browse a website portal. This project was undertaken to design modified version of the Android operating system application and website in which it supports access control policies. These policies limit applications' access to specific users' data and/or resources based on the user's context. The restrictions specified in the policy are applied after studying the behavior of users and the way they interact with our proposed program. Users' information is then saved in the cloud and through that data we can view the registration information and log files. Machine learning can be used to learn the access control rules and distinguish between regular and anomalous access based on the generated dataset. For future work, we are planning to collect more data to refine of results. We also might improve the machine learning schema with some dimensional redaction methods such as selecting the most important features or use algorithmic methods like Principal component analysis (PCA) [35]. We might consider finalizing the final product of our project by connecting it with cloud system.

REFERENCES

1. L. Zhou, C. Su, Z. Li, Z. Liu, and G. P. Hancke, "Automatic fine-grained access control in scada by machine learning," Future Generation Computer Systems, vol. 93, pp. 548–559, 2019. [CrossRef]
2. T. Kalbarczyk, C. Liu, J. Hua, and C. Julien, "Lad: Learning access control polices and detecting access anomalies in smart environments," in 2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS). IEEE, 2019, pp. 485–493. [CrossRef]
3. B. Bezawada, K. Haefner, and I. Ray, "Securing home iot environments with attribute-based access control," in Proceedings of the Third ACM Workshop on Attribute-Based Access Control, 2018, pp. 43–53. [CrossRef]
4. Q. Zhang, S. Wang, D. Zhang, J. Wang, and Y. Zhang, "Time and attribute based dual access control and data integrity verifiable scheme in cloud computing applications," IEEE Access, vol. 7, pp. 137 594–137 607, 2019. [CrossRef]
5. P. M. Khilar, V. Chaudhari, and R. R. Swain, "Trust-based access control in cloud computing using machine learning," in Cloud Computing for Geospatial Big Data Analytics. Springer, 2019, pp. 55–79. [CrossRef]
6. Z. Pan, C.-N. Yang, V. S. Sheng, N. Xiong, and W. Meng, "Machine learning for wireless multimedia data security," 2019. [CrossRef]
7. M. A. A. Khan, M. H. Ali, A. F. Haque, F. Sharmin, and M. I. Jabiullah, "Iot-nfc controlled remote access security and an exploration through machine learning," in 2020 18th International Conference on ICT and Knowledge Engineering (ICT&KE). IEEE, 2020, pp. 1–10. [CrossRef]
8. A. Liu, X. Du, and N. Wang, "Efficient access control permission decision engine based on machine learning," Security and Communication Networks, vol. 2021, 2021. [CrossRef]
9. B. Shebaro, O. Oluwatimi, and E. Bertino, "Context-based access control systems for mobile devices," IEEE Transactions on Dependable and Secure Computing, vol. 12, no. 2, pp. 150–163, 2014. [CrossRef]
10. H. Idrissi, M. Ennahbaoui, E. M. Souidi, A. Revel, and S. Elhajji, "Access control using mobile agents," in 2014 International Conference on Multimedia Computing and Systems (ICMCS). IEEE, 2014, pp. 1216–1221. [CrossRef]
11. V. Chang and M. Ramachandran, "Towards achieving data security with the cloud computing adoption framework," IEEE Transactions on services computing, vol. 9, no. 1, pp. 138–151, 2015. [CrossRef]
12. Y. Hou, S. Garg, L. Hui, D. N. K. Jayakody, R. Jin, and M. S. Hossain, "A data security enhanced access control mechanism in mobile edge computing," IEEE Access, vol. 8, pp. 136 119–136 130, 2020. [CrossRef]
13. L. Liu, E. Yu, and J. Mylopoulos, "Security and privacy requirements analysis within a social setting," in Proceedings. 11th IEEE International Requirements Engineering Conference, 2003. IEEE, 2003, pp. 151–161.
14. J. Liu and J. Yu, "Research on development of android applications," in 2011 4th International Conference on Intelligent Networks and Intelligent Systems. IEEE, 2011, pp. 69–72. [CrossRef]
15. D. F. Williamson, R. A. Parker, and J. S. Kendrick, "The box plot: a simple visual method to interpret data," Annals of internal medicine, vol. 110, no. 11, pp. 916–921, 1989. [CrossRef]
16. M. Frigge, D. C. Hoaglin, and B. Iglewicz, "Some implementations of the boxplot," The American Statistician, vol. 43, no. 1, pp. 50–54, 1989. [CrossRef]
17. J. H. Steiger, "Tests for comparing elements of a correlation matrix." Psychological bulletin, vol. 87, no. 2, p. 245, 1980. [CrossRef]
18. C. D. Dziuban and E. C. Shirkey, "When is a correlation matrix appropriate for factor analysis? some decision rules." Psychological bulletin, vol. 81, no. 6, p. 358, 1974. [CrossRef]
19. T. Neideen and K. Brasel, "Understanding statistical tests," Journal of surgical education, vol. 64, no. 2, pp. 93–96, 2007. [CrossRef]
20. R. Quinlan, C4.5: Programs for Machine Learning. San Mateo, CA: Morgan Kaufmann Publishers, 1993.
21. A. V. Solanki et al., "Data mining techniques using weka classification for sickle cell disease," International Journal of Computer Science and Information Technologies, vol. 5, no. 4, pp. 5857–5860, 2014.
22. N. Saravana and D. V. Gayathri, "Performance and classification evaluation of j48 algorithm and kendall's based j48 algorithm (knj48)," Int. J. Comput. Trends Technol.(IJCTT)–Volume, vol. 59, 2018. [CrossRef]
23. M. N. Amin and M. A. Habib, "Comparison of different classification techniques using weka for hematological data," American Journal of Engineering Research, vol. 4, no. 3, pp. 55–61, 2015.
24. B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," Journal of Applied Science and Technology Trends, vol. 2, no. 01, pp. 20–28, 2021. [CrossRef]
25. H. Berger, D. Merkl, and M. Dittenbach, "Exploiting partial decision trees for feature subset selection in e-mail categorization," in Proceedings of the 2006 ACM symposium on Applied computing, 2006, pp. 1105–1109. [CrossRef]
26. E. Frank and I. H. Witten, "Generating accurate rule sets without global optimization," 1998.
27. A. Kia, P. Timsina, H. N. Joshi, E. Klang, R. R. Gupta, R. M. Freeman, D. L. Reich, M. S. Tomlinson, J. T. Dudley, R. Kohli-Seth et al., "Mews++: enhancing the prediction of clinical deterioration in admitted patients through a machine learning model," Journal of clinical medicine, vol. 9, no. 2, p. 343, 2020. [CrossRef]
28. A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited," in Australasian Joint Conference on Artificial Intelligence. Springer, 2004, pp. 488–499. [CrossRef]
29. H. Langseth and T. D. Nielsen, "Classification using hierarchical naive bayes models," Machine learning, vol. 63, no. 2, pp. 135–159, 2006. [CrossRef]
30. S. Jayachitra and A. Prasanth, "Multi-feature analysis for automated brain stroke classification using weighted gaussian na¨ıve bayes classifier," Journal of Circuits, Systems and Computers, vol. 30, no. 10, p. 2150178, 2021. [CrossRef]
31. G. Holmes, A. Donkin, and I. H. Witten, "Weka: A machine learn- ing workbench," in Proceedings of ANZIIS'94-Australian New Zealnd Intelligent Information Systems Conference. IEEE, 1994, pp. 357–361.
32. M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," International journal of data mining & knowledge management process, vol. 5, no. 2, p. 1, 2015. [CrossRef]
33. P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-validation." Encyclopedia of database systems, vol. 5, pp. 532–538, 2009. [CrossRef]
34. M. H. Zweig and G. Campbell, "Receiver-operating characteristic (roc) plots: a fundamental evaluation tool in clinical medicine," Clinical chemistry, vol. 39, no. 4, pp. 561–577, 1993. [CrossRef]

35. H. Abdi and L. J. Williams, “Principal component analysis,” Wiley interdisciplinary reviews: computational statistics, vol. 2, no. 4, pp. 433–459, 2010. [CrossRef]

AUTHORS PROFILE

Arwa Aloqbi received the B.S. degree in computer Engineer from Taluk University, in 2015, and is currently pursuing the M.S. degrees in information security.

Manal Alshammari is currently pursuing the M.S. degrees in information security.

Amal Alatawi Alshammari is currently pursuing the M.S. degrees in information security.

Adel r. Alharbi, received the B.S. degree in computer science from Qassim University, in 2008, and the two M.S. degrees in security engineering and computer engineering and the Ph.D. degree in computer engineering from Southern Methodist University, in 2013, 2015, and 2017, respectively. He has been a Faculty Member with the College of Computer Science and Information Technology, University of Tabuk, Saudi Arabia, since 2009. His research interest includes mobile and smart device security

Amer Aljaedi, received the B.S. degree from King Saud University, in 2007, the M.S. degree in information systems security from the Concordia University of Edmonton, in 2011, and the Ph.D. degree in security engineering from the University of Colorado at Colorado Springs, Colorado Springs, CO, USA, in 2018. He is currently an Assistant Professor with the College of Computing and Information Technology, University of Tabuk. His research interests include broadly in SDN, network traffic control and monitoring, and cybersecurity

.

.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)