ISSN 2278-3091

Volume 9, No.1.1, 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse5191.12020.pdf

https://doi.org/10.30534/ijatcse/2020/5191.12020

Authentication and Authorization Model for Safeguarding Mission Critical Applications in Pervasive Environment

Norhusna Binti Baharom[1], Manmeet Mahinderjit Singh [2]

[1]School of Computer Sciences, Universiti Sains Malaysia, Penang, nhusnas.ucom13@student.usm.my

[2] School of Computer Sciences, Universiti Sains Malaysia, Penang, manmeet@usm.my

ABSTRACT

Smart city applications nowadays consist of mission-critical applications such as Mobile Crowdsensing (MCS), Internet of Things and Fourth Industrial Revolution (4IR) based manufacturing applications. Sensing data are shared automatically by all the mission-critical system causing challenges of data leakage leading to loss of user privacy. This study aims to verify the security vulnerability in one of the types of mission-critical system which is the MCS application. The vulnerability will be evaluated based on the two risk-assessment models which are the CIA Model and AAA Model. An essential section of the paper will focus on the proposed authentications and authorization authentication model for safeguarding and protecting any types of mission-critical applications.

Key words : Mobile Applications , Industrial Revolutions, Access Control, Adaptive Risk Model, Direct Assertion Identity

1. INTRODUCTION

Mobile applications are becoming more popular these days. The fact that we can just open an application to perform any specific task is quite useful in our everyday lives. A mobile application that take advantage of sensors functionality and runs on pervasive environments are on par with a few technological trends such as Big Data, Internet of Thing (IoT), Mobile Crowdsensing (MCS) and Fourth Industrial revolution (4IR). All these groups of applications fall under mission critical applications and system based. In this study, two parts of sharing in term of security vulnerabilities of one type of mission critical application which is MCS will be presented. The second part aim to present a comprehensive authentication protection for any mission critical applications in pervasive environment.

MCS applications can be divided into three types of crowdsensing application which are environmental sensing (pollution), infrastructure sensing (restaurants) and the social sensing (etc. Twitter). The MCS application is classified into either Participatory Sensing and Opportunistic Sensing. Participatory sensing is where users control the information they shared while Opportunistic Sensing data are shared automatically by the system. Both sensing applications have their own pros and cons. Let's look at this scenario: Global Positioning System (GPS) sensor in mobile application, for example – Previously, before this kind of applications were created, whenever a person needs to go to a destination that is new to him or her, the person had to rely on the direction sign board, solely. And probably, ask the locals about the direction to the destination. And there will also be times where even the locals do not know the direction of the destination. This eventually leads to time-wasting if he or she has taken the unnecessary routes. Fortunately, nowadays, developers have programmed many types of MCS application where each type of MCS application has distinct functionalities. Thus, the previous scenario can be solved by using MCS application that consists of the GPS sensor. This application will enable users to input their destinations and it will guide them with the right way to reach the designated destination. Unfortunately, like in every application that involves a distributed network, the security factors will always become an issue for both developers and users. Some of the findings do not even have a solid proof to support their methods. With current findings, the methodology has not been proven with real life application. Security issues on Mobile Crowdsensing applications raise a few questions for future development. For example, what are the security vulnerabilities or weaknesses of mobile crowdsensing applications concerning the authentication issues? What is the authentication security measures that can be done to mitigate the occurrence of the security attacks in a mobile crowdsensing application? Therefore, this research will focus on verifying security vulnerability study of the mobile crowdsensing applications to answer the raised questions. Consequently, based on the findings, a comprehensive authentication and authorization factors pertaining to mission critical applications will be presented.

The outline of the paper is as the following. Section 2 outline the research background. Section 3 and Section 4 are respectively on analysis of MCS application and its results. Section 5 present a comprehensive discussion on the MCS authentication safeguarding mechanism. Finally, a conclusion section is presented.

2. RELATED WORK

In this section, the literature background will be discussed in depth.

Table 1: Existing MCS Applications

Application	Description	Type of MCS sensing	MCS application Phenomenon
CreekWatch[4-5]	iPhone application that monitors water level	Participatory	Environmental
Nericell [6]	Monitors the traffic.	Opportunistic	Infrastructure
DietSense [4]	Users can share their eating habits.	Participatory	Social
Amazon's Mecanical Turk (MTurk) [7]	It is an Internet marketplace.	Both	Social
Waze [8]	Provides personalized route direction.	Both	Infrastructure & Social
CarTel [9-10]	It is installed in cars to measure the location and the speed of the car.	Both	Infrastructure
BikeNet [3,11]	Monitors bike routes and the condition of the route.	Both	Infrastructure & Social
PIER[12]	Monitors the environment from user's location data.	Participatory	Infrastructure
N-Smarts[13]	Monitors pollution.	Opportunistic	Environmental
Common Sense[3,14]	Uses handheld air quality sensing device that communicates with mobile phones.	Opportunistic	Environmental

Nowadays, our community depends on mobile devices in most of the task they execute in everyday lives. One of the most important features in a mobile device is the ability for it to efficiently run the Mobile Crowdsensing (MCS) applications. MCS is an application that influences user's mobile devices (GPS, smart phones, car sensors) to gather and distribute data about the user either interactively or autonomously towards a common goal [1]. According to [2], a MCS application is a mobile application that includes various types of mobile sensors such as camera, microphone, GPS which is able to sense the user's data. Mobile Crowdsensing applications can help us in many ways, such as monitoring the environment (pollution), transportation/traffic planning (Waze) and mobile social recommendation (TripAdvisor). MCS applications can be widely grouped into two sensing classes which is personal sensing and community sensing. The sensor that focuses on the experience related to that one person such as observing the movement patterns (e.g. jogging, sitting, and walking) of oneself is known as the personal sensing application. The type of sensor that involves in observing a bigger scalar of phenomena which needs to include the participation of several individuals to provide proper results such as monitoring the traffic jam is known as the community sensing application [3]. There are various types of existing Mobile Crowdsensing applications that is available for a public use. Table 1 shows the MCS applications that were proposed by previous researchers.

Each of the MCS applications that use the sensors can contribute to different purposes. Let's take GPS (along with a few other sensors) for example, in [6] it is used to detect potholes, bumps, braking and honking while in [8], it provides a personalized route directions. This shows that technologies can be adapted and enhanced, where the function of the sensors was improved according to the needs of today's generation. There are many challenges in the Mobile Crowdsensing. Data reliability issues caused by the submission of false data by users will affect the results critically. The heterogeneity of mobile platforms (Android, IOS, etc.) can also become a challenge to the MCS Application as it needs a different application development. A high network bandwidth utilization is a challenge because the core network will get congested – For example; video sharing. Past research proposed having a primitive processing of raw data on the device to be performed by some local analytics in regards of the localized analytics issues [6]. CarTel was proposed with a method that prioritized the task of data collection to overcome the resource limitations problem [10]. Anonymity has become one of the privacy, security and data integrity solution [15].

OWASP is an online community that provides freely available articles, methodologies, documentation, tools and technologies in the web application security field. It created its own risk modelling and the top ten risks in mobile [16].

2.1 Current Security Attacks on MCS Applications

There are various kinds of attacks happens in MCS applications. Generally, it is divided into two types of attacks which is a Passive Attack and an Active Attack. A passive attack usually keeps the information of other user's data and does not cause any damage to the system. Eavesdropping is an example of a passive attack. An active attack involves in modifying user's information in some way that can cause damage to both the system and the targeted user's information. The main three attacks are as following. Man-in-the-Middle Attack (MiMA) is a form of eavesdropping where interactions between two users is monitored and altered by an unauthorized party. Generally, the attacker interrupts and changes the network traffic of the devices between two targeted users that communicates with each other. Basically, the attacker will intercept a public key message exchange and retransmit the message while replacing the requested key with his own [17]. A previous study [18] proposed a method where the service provider on a client's system will provide a secure service for each client named the spy. The server will runs a protocol that validates the integrity of the client with the help of the spy. If the client's integrity cannot be validated, it will deny the service provision to the client. Besides that, MiMA can be mitigated by implementing authentication, [19] proposed a remote autonomous object with public key cryptosystem of the user's authentication that can be implemented in telecommunication system called Navikov-Kiselev scheme.

The security attack that could violate the system's availability is the Denial of Service (DoS) attack. DoS is an attack against a computer that provides services to the customers over the Internet. It counters a valid user from accessing the information or services. Typically, DoS is done by flooding the targeted network/machine with excessive requests to overload the system. This will eventually cause the honest users unable to get their legitimate requests from being executed [20]. A simple way of explaining this concept is by imagining that if a door of a building is full of a group of people and the workers of the building cannot enter the building which will lead to disruption of the normal operations. These attacks are considered as the most dangerous, but the least likely to happen. If the attacks were applied to a professional setting: even simple DoS could prevent doctors from treating their patients correctly or halt a law enforcement agent from gaining information to catch lawbreakers. [21] states that one method in preventing a DoS attack is by giving only the authorized node to exchange the routing information and the cryptographic-authentication mechanism. The eHealth [22] also consider DoS attack in designing their method.

Sybil Attack. In social networking, users can create their personal account, share information and engage in a vast network of friends that will include a lot of strangers. Having to handle a wide variety of features such as games, puzzles, real time applications, image sharing, instant messaging, and so on, the personal information shared amongst a group can be exposed. A Sybil attack can alter the overall ranking in a voting application, bad-mouth an opinion, access resources or breaking the mechanism of trust behind a Peer-to-Peer (P2P) network [23,24,25]. It is said that attacks on direct nodes in a distributed network are easier, but harder to detect [26-27]. The open architecture of the social networking is the main reason for the attacks to happen. Due to this, even the usual culprits like spams, cross scripting, and social engineering can cause great damage. The consequences of the Sybil attack vary based on either the attacker is inside or outside of the distributed network. In Sybil attack, the malicious identity can also create many forged identities to acquire user's sensitive data. If the enemy has at least one forged identity and is a part of the network, then she or he is an Insider, otherwise the attacker is an Outsider. An Insider can present numbers of forged identities and act as a valid identity. The enemy can forge all Sybil nodes in the same time or unveiling them one by one [27]. The Sybil attack can also jeopardize the whole network. A past study [28] had proposed a method where they determine the fake accounts on the network by using data mining on peer's profiles of the victim. There were many past studies addressing the privacy and security of the sensed data in the MCS application [29-31]. The most common solution is by implementing the anonymization [15] by removing private data before sharing it. However, even though anonymization increases data privacy, it reduces the usefulness of the data. And sometimes even if we have removed our name or address, it cannot guarantee the anonymity of users such as collecting location data. Having access to the user's context, a person could collect more information about that person. This is worrisome as an accurate user profiling will be enabled. Usually, the data might be at risk to be accessed by the third parties either during the reporting, the processing or the storing of data [1,30].

2.2 MCS and Smart City

MCS and Smart City are linked closely together. Mobile crowdsensing paradigms is an important source of sensor data sharing community of many mobile devices carried by people around the world to support their daily lives while Smart City incorporates a data sharing option of a set of applications. Nowadays, cities are facing a complex challenge to satisfy the aim of the socioeconomic development and the quality of life. As a result, the 'Smart City' concept was introduced as it is considered as a promising resolution by giving a productive service to the civilians with the employment of Information and Communication Technologies [32]. It aims to enhance citizen's life value and gives continuous services [30]. With the expected population of 2.3 billion in the next forty years, with estimated 70% of world population lives in the city [33], Smart City will be able to provide a better quality of life to the citizens.

As the popularity of the use of a mobile social networking escalated, MCS has been viewed as a competent resolution to some issues concerning the collection of data involving many partakers. In other words, MCS applications can be utilized to develop a Smart City application [34]. The Smart City concept incorporates moving sensors and human intelligent into the sensing process with the addition of sensor network technology. In the process of implementing a Smart City application, one of the steps is developing applications that accommodate the Smart City requirements with the help of the MCS [33]. Smart City applications depend on gathering accessible information from sensor networks and form an intelligent service application.

A past research on Smart Parking [34] attempt a specific case study on whether the basic design principle is applicable to similar applications array, they design a Smart Parking system that utilizes the MCS concept. They proposed a 'coordinated crowdsensing' where the system integrates information collected from users and assist them to examine the unknown area. Next, their solution implements a sophisticated data collection mechanism in the case of inadequate data and participants. They also study on tolerating free riders; users who use the application but did not contribute any new information and could be causing the decreasing quality of the service. Based on the findings, free riders can exist as long as there is a sufficient number of contributors in the system.

2.3 Authentication Techniques against MiMA

MiMA exploits the weaknesses of authentication protocols that is being used by the communicating parties. The system certificate generation is another possible weakness as authentication certificates are commonly given by the third parties. Referring to the AAA Model, authentication factor is one of the elements in the model. Authentication is a security measure designed to verify and validate user's or station's credentials before giving them the access to the resources. Authentication is classified into two which are: Single-factor authentication (SFA) and Multi-factor authentication (MFA). SFA depends on how attentive of the user to take extra measures [35]. The most secure method that can be done in SFA is by creating a strong password that will usually include alphabets, numbers and the special characters. However, the problem that might occur is that people tend to set the same password combination across multiple sites. A Domino effect will occur if an attacker managed to get their password.

Nowadays, observing recent authentication trends, the popularity of a multi-factor authentication has widely escalated within the field of systems and applications. The framework of the mechanism is more secure compared to a single-factor authentication method by providing a dual means of verifying credentials from separate category. Some example of applications or systems that uses MFA are Google account, Yahoo account, Linkedin account, and Twitter account [36]. The main benefit of MFA is that it enables a stronger security mechanism by asking users to give more than one identification prove from separate categories of authentication factors. The Two-Factor Authentication (TFA) is a type of MFA.

Based on the study done, there are three types of user authentication implemented in mobile applications which is the password authentication, smart card authentication and the biometric authentication. Majority of MCS applications uses the simple password authentication [Google Play] where the system user will have an ID and password in order to access the application. Unfortunately, this method is the least secure method compared to the other two [35,38]. Hence, a more secure password method was developed and known as the One Time Password (OTP) and one of the methods was proposed by Raihi et al. [37,16]. The system will generate OTP passcode randomly as the replacement of the password that needs the user to remember it.

As been mentioned in [38], a Two-Factor Authentication (TFA) is a type of MFA where it includes what you know and what you have. By that it means, the password that you know and a device that you have. TFA requires users to prove themselves in two different ways before you can access the service of the application. Many mobile applications nowadays use One Time Password (OTP) in their TFA [36]. Unlike the static secret code like your ATM card, the OTP changes every time users request a login session. In theory, the attacker will have a hard time stealing user's information as the password they managed to hack alone might not be enough to access the next authentication layer. There are a few types of TFA:

i. OTP via SMS

This method is where users need to enter an additional passcode after they have authenticated themselves with their username and password. It is inexpensive and easy to implement. A text message (SMS) that contains the passcode will be send to user's mobile device. After the user received the code, they will have to enter the code into the running application to proceed with the login. But unfortunately, the SMS code does not work for everyone. This is because the server might not have registered the user's country in their list of supported regions and cannot send the code to the user. Besides that, the OTP needs mobile coverage to send users the code or else it would not be able to send the code. More importantly, the adversary may intercept user's mobile line to get the code sent [39].

ii. Authenticator Apps

Authenticator apps use the same concept as the OTP via SMS. But instead of having the passcode being sent to the phone via SMS, the passcode in thimethod will be generated locally within the mobile phone itself. This type of authentication relies on the cryptographic algorithms for Time-based One Time Password (TOTP). The passcode will be cryptographically generated from the secret starting key or a 'seed' that is stored by the server by considering the current time and the date when the user login. This passcode will be valid for approximately 30 to 60 seconds. Usually, the 'seed' will be sent to the user via QR code and then it will be imported into the authenticator app that the user is using. The only thing the user needs to keep in mind is that the date and time on the device must be accurate to get the correct passcode. With this method, even though the enemy managed to get thousands of user's password records, they would not be able to figure out the sequence and what might be the passcode next unless they manage to get the 'seed'. One example of an authenticator app is the Google Authenticator [40].

iii. Biometric Recognition

Biometric authentication relies on something that you are. For examples, fingerprints, face, iris and voice recognition. Theoretically, this type of authentication sure is the most secure as it uses a unique feature that a person have. Unfortunately, with it carrying such role, the obstacles of implementing the method will be quite challenging. Let us take the registration process as an example. In order to implement the biometric, the environment surroundings must be considered. If the user wants to register his/her voice, then his/her surroundings must be noise-less for the app to capture the accurate features of the voice. This does not only apply to the registration but also to includes the verification step because even if the user managed to register his/her voice, a slight change to the voice features (such as having flu) might affects the verification process. However, it is not implemented much in MCS apps due to the complexity of its implementation. Fortunately, a study by Hoang et al. has founded that the biometric authentication can be implemented into a mobile application [41,42].

Considering the methods studied, the authenticator apps would be a suitable choice for implementing the TFA in the proof of concept app.

3. SECURITY ANALYSIS FOR MCS APPLICATIONS

In order to determine the security vulnerability of a mobile application that might be accessible to the security attacks, a security tool named Burp Suite has been selected. In this case, Man in the Middle attack will be used as a case study. Burp Suite will be testing out the application layer of the mobile application for its security vulnerability. The aim of the tool is to act as the Man in the Middle and aid in collecting more accurate information for that particular session of the testing process.

Ten Mobile Crowd Sensing applications have been selected from Google Play to undergo the vulnerability testing. These mobile applications will then be observed for its features. The features of the applications that were considered are the type, the name, the main functionalities, the sensors, the presence of authentication mechanism, and the need of the application to be connected to the internet. The information about the applications is shown in Table 2.

Table 2: Mobile Crowdsensing Applications Selected for Pen testing Vulnerability test

Application Type	Application Name	Functionalities	Sensors	Authentication Mechanism	Internet Connection
Smart parking Apps	RTA Smart Parking (3.1)	Find free parking spots and get their respective tariff and also pay parking fee.	GPS	Cleartext Password	Yes
	Free Parking (1.5.4)	Finds the nearest free parking around the user.	GPS	Absent	Yes
	Find My Car: Parking reminder (1.6)	Finding the location of your parked car.	GPS	Absent	Yes
Noise Pollution Apps	NoiseTube Mobile (2.0.2)	Measures environmental noise of a city and represents results as maps.	Wireless Cellular Network	Cleartext Password	Yes
Environmental Apps	Clean Air Make More (2.4)	Provides real-time information about the quality of air.	Wireless Cellular Network	Absent	Yes
	Clean City Network (1.0.15)	Uses M2M technology that can monitor real-time waste bin level.	GPS, Accelerometer and External sensor	Password encryption	Yes
Smart Road Navigation Apps	Waze (4.9.0.2)	Community-based traffic and navigation application for drivers around a certain geographical area.	GPS and Camera	One-Factor Authentication: Phone No. & One Time Password	Yes
	Here WeGo (2.0.10751)	City navigation with bike, car, taxi and more	GPS and Accelerometer	Cleartext Password	Yes
	Traffic Authority (1.1)	Provides up-to-date information about traffic conditions	Wireless Cellular Network	Absent	Yes
Smart Road (Road Bump and Pothole monitoring) Apps	Road BUMP (1.3.7)	Used to upload exact location of potholes on a street so that the appropriate authority can respond and fill up the potholes.	Wireless Cellular Network	Cleartext Password	Yes

The ten selected applications are from a few types of Smart City application which are the Smart Parking, Noise Pollution, Environmental, Smart Road Navigation, and the Smart Road Bump and Pothole Monitoring. Based on the table above, the sensors used differ according to their functionalities. Besides that, majority of the tested apps have

their own authentication mechanism. We can also see that 100% of the selected apps needed the Internet connection.

In order to identify the vulnerability that is commonly happens in a Mobile Crowd Sensing application concerning the Man in the Middle attack, an observation of the security problems and the analysis of it have been conducted. For the observation, the author used a security tool called Burp Suite.

Burp Suite is a tool that can track the activity of the sensors used by the user through their mobile devices. The three main processes of the testing are as shown in the flow chart in Fig.1.

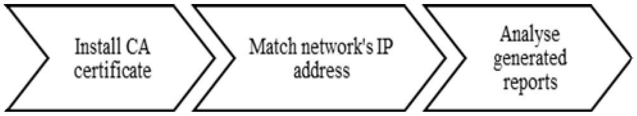

Figure 1: Testing flowchart

As to ensure that no unnecessary apps or ads disturbing the testing process, the phone will be rooted beforehand. First step is the certificate authority (CA certificate) installation. CA is an entity that oversees giving out the digital certificate. This digital certificate act as a signature that will verify the owner of a public key based on the certificate's named subject. Basically, with the CA, it will be able to verify whether the third party can be trusted or not and hence, authorizes the communication between the 2 entity's identities on the Internet. In our everyday lives, a CA can be installed in various ways. One of it is by clicking the link attached in emails that contains the CA and it will be automatically installed to the device. In this case, the author installed the CA manually. The mobile phone will be installed with a CA that matches with the Burp Suite's CA certificate. The next step is matching the IP address of the phone's network to the IP address of the laptop's network. This is done by changing the IP address of the mobile phone so that it matches the IP address of the Burp Suite (laptop). To get the IP address from the laptop, you can search 'Command Prompt'. Then prompt the command 'ipconfig' for Windows and 'ifconfig' for a Linux Operating System. The Burp Suite will be able to track and intercept the user's communication with the server. Each request and response between the server and the user will be exposed to the 'enemy'.Lastly, as the connection between the mobile phone and the Burp Suite has been established, the Burp Suite will start generating dynamic analysis reports based on the phone's activity. In order to evaluate the applications, we need to 'play-around' with the application one at a time. Some information that we managed to extract from the report are the login credentials, location, time, internet protocol and device's information.Table 3 above shows the exposed information generated by the Burp Suite. The full analyzed report will be included in the Appendix section. Table 4 below shows the analyses of the arranged reports.

Table 3: Example of generated reports by Burp Suite

Exposed element	Generated Analysis
Protocol	HTTP/1.1
User's Device	User-Agent: Mozilla/5.0 (Linux; Android 5.0.1; GT-I9500 Version/4.0 Chrome/54.0.2840.85
Username & Password (UN & PW)	j_username=dhalynaadam%40yahoo.com&j_password=v81U496753Q
Date & Time	Date: Fri, 02 Dec 2016 04:40:52 GMT
User's Information	{"userId":"HERE-4768e5a1-950f-4cf5-8e6a-66f01bf17155","firstname":"First","lastname":"Last","email":"dhalynaadam@yahoo.com","dob":"17/12/1997","language":"en","countryCode":"MYS","emailVerified":false,"marketingEnabled":false,"createdTime":1481961689925,"updatedTime":1481961689901,"state":"enabled"}
Cookie	Set-Cookie: PLAY_ACTIVE_ACCOUNT=ICrt_XL61NBE_S0rhk8RpG0k65e0XwQVdDlvB6kxiQ8=nsikakowoh12@gmail.com;Path=/

4. SECURITY ANALYSIS RESULTS

The report of the request and response of the user and the server respectively for each application were analyzed. From the report of the applications generated, we managed to extract the exposed information, the type of authentication, and the information that are encrypted. Example of the exposed information in the generated reports and presented in Table 4.

Table 4: Exposed Information Captured for Mobile Crowdsensing Applications

Smart City Applications	Exposed Information	Authentication mechanism	Encrypted Information
RTA Smart Parking	Protocol, User’s Device, UN & PW, Date	Password : Email & PW	Cookie
Waze	Protocol, User’s Device	2 Factor : Number phone & OTP	Cookie
Clean City Networks	Protocol, User’s Device, Date, UN	Password : Email & PW	Password
Noisetube	Protocol, User’s Device, User’s Information, Date	Password : Username & PW	Cookie
Free Parking	Protocol, User’s Device, Date	-	Cookie
Road BUMP	Protocol, Date, UN & PW	Password : Username & PW	-
Here WeGo	Protocol, User’s Device, Date, User’s Information	Password : Username & PW	-
Traffic Authority	Protocol, User’s Device, Date	-	Cookie
Clear air Make More	Protocol, User’s Device, Date	-	-
Find My Car: Parking reminder	Protocol, User’s Device, Date, Cookie	-	Cookie

The test was mainly focused on retrieving the login credentials of users to prove that user’s data can be vulnerable to the sniffers. The vulnerability of each application differs based on the architecture of the applications. Some apps are more vulnerable compared to the others. Based on the table above, we can see that 90% of the selected apps are vulnerable to the Man in the Middle attack to a certain extent as user’s information was exposed. The internet protocol the app used, time, date, location, device information was exposed for majority of the apps. The one app that managed to protect their user’s information from Burp Suite is the ‘Waze: Navigation app’. Compared to other applications, Waze uses a multi-factor authentication as their authentication mechanism where users need to enter their phone number and then the application will send a One Time Password (OTP) to the user’s phone. 60% of the session cookies of these apps were encrypted. This adds on another layer of protection to the user data. Approximately 83% of the app that implemented authentication mechanism used a simple password authentication. This causes the enemy to easily hack into user’s information.

5. AUTHENTICATION MECHANISMS FINDINGS AND DISCUSSION

From the test conducted, Man in the Middle managed to intercept 100% of the user’s information to some extent. The reason of this is that the communication channel is prone to this type of attack. This is also backed with the work done by [18]. Consider any banking apps account for an example. Nowadays, technologies enable us to do almost everything just by having a smart phone. This also includes the online banking. Imagine if you sent your phone for a fix at some phone-fixing shop. If the fixer is a ‘bad’ person, they will install the fake CA to your phone and they will be able to see your activity. Even if your cookie session were encrypted, they still can do a dictionary attack to decrypt the code. Overall, majority of the application shows that authentication is the main issue. This is also in line with OWASP where they state that authentication problem happens a lot in mobile applications [16]. Next a comprehensive discussion on safeguarding authentications revolving MCS will be presented. Authentication remains as the first defense in any mission critical applications such as IoT, MCS and even fourth Industrial Revolution(4IR) based applications and among the protection mechanism involves with adopting multifactor authentications integrated with strong hashing algorithms besides encrypting other essential information's. In practice and as shown in Fig 2, mission critical applications involves three main stakeholders which are the mobile users, applications and Application servers. The main essential in any authentication models originating from IoT paradigm such as MCS and 4IR based applications is by tapping on security by design model. Each pervasive application differs based on its functionality and its design and requirements. Similarly, each application and pervasive domain has different types of system vulnerabilities and threats. The need to provides safeguards measurement based on different applications and domain types to embed security safeguards at its design stage is important. Next, for any mission critical applications and domain, the need to have authentication protection for identifying human to human, and device-to-human communications is essential. Finally, any metadata within the application should be protected based on its sensitivity. Each sensors data captured should be automatic annotated based on its sensitivity [46]. Sensors such as GPS which stands as a sensor that leads to leakage of one's locations information's has a high sensitivity in contrast to accelerometer sensor [46]. This information is then could be inputted as in designing access control in terms of adding user permission and its level. Another promising path is in protection of metadata using technologies which are secure and tamper proof such as blockchain technologies [47]. The mechanisms of authentication are divided based on authenticator IDS, channel/communications, protocol and technologies and authentication requirements. Figure 2 display Authentication factors towards MCS Applications in Smart Cities. Next, the we will present the four-authentication mechanism presented.

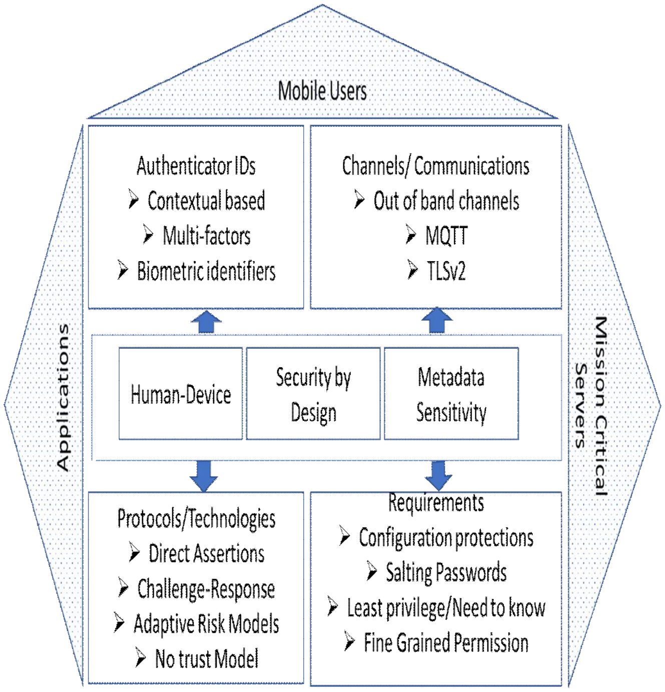

Figure 2 : Authentication and Authorization Model Safeguarding Mission Critical Applications

i) Authenticator IDs

The identifiers or features that become input to identify a user has emerged beyond simple text-based passwords. With the latest technologies' advancement, the need to use biometric based identifiers are becoming acceptable and practical. This is because biometrics are unique and cannot be masqueraded. Biometrics identifiers adopted here could originated from hard (fingerprint, face) and soft (keystrokes, voice) identifiers. When a single biometric or unimodal biometric identifier are integrated with other factors such as password, token or proximity sensors technologies (NFC, RFID), the outcome leads to a multi-factor-based system. Currently, framework such as FIDO [49] employs multifactor identifiers with a list of biometrics factors and sensor-based factor. Nevertheless, the factors mentioned are also capable to be merged with contextual factors (time, location, behavior and user ID) to support applications and systems based on sentient environment [48]. Applications protected and accessible based on contextual information's are important to permit authorized user access during certain time and limiting full access within restricted perimeter.

ii) Channel/Communications

Most of the pervasive applications stored and used online are communicated through unprotected HTTP browser. The need to protect the communication between stakeholders such as mobile users accessing applications with the applications server need to be done using channel such as MQTT and SSL/TLSv2. Another promising method of authentication which is adopted in wearable technologies is by employing out of band channels. Here two different frequency bands such as WIFI used by Smartphone communicating with broadband provider and Bluetooth channel used between smartphone and wearable device. In term of security, this kind of authentication is prone to inherit all the vulnerabilities of WIFI and Bluetooth. But the positive sides is the flexibility of adding security protection device such as Smartphone to protect wearable devices.

iii) Authentications Protocol and Technologies

In any mission critical based applications, the need to adopt an adaptive risk model has become a major importance. Any system should be repeated accessed in detecting new threats and new vulnerabilities to eliminate the chances to be attacked by hackers. With all authentication applications adopting multi-factors capabilities, another protocol must be adopted is a challenge response protocol. The usage of One-time password (OTP) within system ensure timeliness and most of all thwarts against most of security attacks such as phishing attack, password attack and even fraud revolving financial systems. Challenge-response is also a key in handling attacks of ecommerce transaction attacks. Another important protocol needs to be tapped into any applications is the direct assertions. As most of the authorization permission are given right after user authenticate themselves, this could lead to issue of an unauthorized user still manage to brute force and penetrate a system. With most of the system defined to allow least privilege, this unauthorized and informed guest can at least read any files. This indirect assertion should not be the correct way to allow users in accessing an application [50] due to its systemic failure leading to data theft loss of almost USD600 million in 2017 itself. The correct way is for all authorization permissions and rights should be accessed for a user before this user is invited to authenticate himself. The usage of direct assertion could be the answer to all the security issues revolving any technology and applications. Finally, with the importance of each devices and users no trusting each other, the need for all connected devices and user to be accessed before the authentication process is essential.

iv) Other Authentication Requirements

Other essential requirements for any applications when it comes to authentication would be to add features of protections in its configuration files. All pervasive applications have a configuration file in which metadata containing details such as mobile standard, types, its version, password of admin, IP address, etc are stored. Normally this configuration files could be accessible by the application users. In an event if the configuration files are accessed by hackers, the leakage of metadata of the applications and even the devices could take place. As one way of safeguarding configuration files is by using shadow file mechanism. In this approach, the configuration files contain pointer to another locations which is accessible only by authorized administrators. This second layer of defense in depth can be further integrated with strong password mechanism by using salting random numbers with user password. With salt being used, the chances of password duplication and brute force of password is minimized. Other requirement is in driving access control system towards only allowing least privilege access and in accordance to need to know principles should be applied. A user is only given access based on their needed tasks with the lowest clearance such as to read. This will also ensure that any permission on object or functionalities of an application is presented in fine grained manner.

Overall, the presented model is capable to secure and safeguards applications from both security and privacy attacks targeting mission critical applications.

6. CONCLUSION

In this paper, an analysis on several MCS applications in a smart city environment is presented. The findings shows almost 70% of applications are not being protected and lack of authentications and encryptions protections. Since authentication stands as the first defense of protections, we have formulated factors revolving both authentication and authorization which are mandatory to protect any MCS applications. The proposed model could also be adopted for any mission critical applications design in a pervasive environment and even on par with the requirements compulsory for 4th Industrial Revolution (4IR) trend.

ACKNOWLEDGEMENT

This work is funded by a Research Grant provided by University Sains Malaysia. The Research Grant title is Ambient Intelligence Smart Home Automation using ECHONET LITE (AmISHA-ELITE). The account number of this grant is 1001/PKOMP/8014003.

REFERENCES

1. Hu, X., Li, X., Ngai, E. C. H., Leung, V. C., & Kruchten, P. (2014). Multidimensional context-aware social network architecture for mobile crowdsensing. IEEE Communications Magazine, 52(6), 78-87. https://doi.org/10.1109/MCOM.2014.6829948

2. Talasila, M., Curtmola, R., & Borcea, C. (2015). Mobile Crowd Sensing.

3. Ganti, R. K., Ye, F., & Lei, H. (2011). Mobile crowdsensing: current state and future challenges. IEEE Communications Magazine, 49(11), 32-39.

4. Dimov, D. (2014). Crowdsensing: State of the Art and Privacy Aspects - InfoSec Resources. Retrieved September 29, 2016, from http://resources.infosecinstitute.com/crowdsensing-state-art-privacy-aspects/

5. IBM Corporation 2010. (2010). Explore your watershed. Retrieved September 29, 2016, from http://creekwatch.researchlabs.ibm.com/

6. Mohan, P., Padmanabhan, V. N., & Ramjee, R. (2008, November). Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In Proceedings of the 6th ACM conference on Embedded network sensor systems, 323-336. ACM. https://doi.org/10.1145/1460412.1460450

7. Amazon Inc. (2005). Amazon Mechanical Turk - Welcome. Retrieved September 23, 2016, from https://www.mturk.com/mturk/welcome

8. Waze Mobile. (2006). Free Community-based Mapping, Traffic & Navigation App. Retrieved September 24, 2016, from https://www.waze.com/

9. CarTel Team. (2009). Traffic Mitigation. Retrieved September 27, 2016, from http://cartel.csail.mit.edu/doku.php

10. Hull, B., Bychkovsky, V., Zhang, Y., Chen, K., Goraczko, M., Miu, A.,Madden, S. (2006). CarTel: a distributed mobile sensor computing system. Proceedings of the 4th International Conference on Embedded Networked Sensor Systems - SenSys '06, 125-138. doi:10.1145/1182807.1182821

11. Eisenman, S. B., Miluzzo, E., Lane, N. D., Peterson, R. A., Ahn, G., & Campbell, A. T. (2007). The BikeNet mobile sensing system for cyclist experience mapping.Proceedings of

the 5th International Conference on Embedded Networked Sensor Systems - SenSys '07, 87-101. doi:10.1145/1322263.1322273

12. Mun, M., Boda, P., Reddy, S., Shilton, K., Yau, N., Burke, J., West, R. (2009). PEIR, the personal environmental impact report, as a platform for participatory sensing systems research. Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services - Mobisys '09, 55-68. doi:10.1145/1555816.1555823

13. Honicky, R., Brewer, E. A., Paulos, E., & White, R. (2008). Nsmarts. Proceedings of the Second ACM SIGCOMM Workshop on Networked Systems for Developing Regions - NSDR '08, 25-30. doi:10.1145/1397705.1397713

14. Afreen, S., Abid, K., & Akula, V. G. (2013). A novel approach on Applications, Research Challenges and Mining for Mobile Crowdsensing. International Journal of Advanced Trends in Computer Science and Engineering, 2(1), 75-80.

15. Sweeney, L. (2002). k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(05), 557-570 https://doi.org/10.1142/S0218488502001648

16. M'raihi, D., Bellare, M., Hoornaert, F., Naccache, D., & Ranen, O. (2005). Hotp: An hmac-based one-time password algorithm (No. RFC 4226).

17. Rouse, M. (2015). Man-in-the-Middle Attack (MitM), Retrieved on October 28,2016, from http://internetofthingsagenda.techtarget.com/definition/man-in-themiddle-attack-MitM

18. Serpanos, D. N., & Lipton, R. J. (2001). Defense against man-in-the-middle attack in client-server systems. In Computers and Communications, 2001. Proceedings. Sixth IEEE Symposium on (pp. 9-14). IEEE.

19. Novikov, S. N., & Kiselev, A. A. (2003, July). The authentication of the user from the remote autonomous object. In Electron Devices and Materials, 2003. Proceedings. 4th Annual 2003 Siberian Russian Workshop on (pp. 137-138). IEEE.

20. McDowell, M. (2013). Understanding Denial-of-Service Attacks, Retrieved on October 29,2016, from https://www.us-cert.gov/ncas/tips/ST04-015

21. Wood, A. D., & Stankovic, J. A. (2002). Denial of service in sensor networks. computer, 35(10), 54-62. https://doi.org/10.1109/MC.2002.1039518

22. Abie, H. and Balasingham, I., 2012, February. Risk-based adaptive security for smart IoT in eHealth. In Proceedings of the 7th International Conference on Body Area Networks (pp. 269-275). ICST (Institute for Computer Sciences, SocialInformatics and Telecommunications Engineering).

23. Rathee, P., & Malhotra, S. (2015). Preventing Sybil Attack in Wireless Sensor Networks. International Journal for Innovative Research in Science and Technology, 1(12), 561-565.

24. Chang, W., & Wu, J. A Survey of Sybil Attacks in Networks. Sensor Networks for Sustainable Development. Proceedings of the 3rd international symposium on Information processing in sensor networks , 259-268.

25. Pawar, S., & Vanwari, P. Sybil Attack in Internet of Things.

26. Rakesh, G. V., Rangaswamy, S., Hegde, V., & Shoba, G. (2014). A Survey of techniques to defend against Sybil attacks in Social Networks. International Journal of Advanced Research in Computer and Communication Engineering, 3(5).

27. Newsome, J., Shi, E., Song, D., & Perrig, A. (2004, April). The sybil attack in sensor networks: analysis & defenses. In Proceedings of the 3rd international symposium on Information processing in sensor networks (pp. 259-268). ACM. https://doi.org/10.1145/984622.984660

28. Jhaveri, H., Jhaveri, H., & Sanghavi, D. (2014). Sybil Attack and its Proposed Solution. International Journal of Computer Applications, 105(3).

29. Gisdakis, S., Giannetsos, T., & Papadimitratos, P. (2016). Security, Privacy, and Incentive Provision for Mobile Crowd Sensing Systems. IEEE Internet of Things Journal, 3(5), 839-853. doi:10.1109/jiot.2016.2560768

30. Cilliers, L., & Flowerday, S. (2014). Information Privacy Concerns in a Participatory Crowdsourcing Smart City Project. Journal of Internet Technology and Secured Transaction, 3(3), 280-287. doi:10.20533/jitst.2046.3723.2014.0036.

31. Gisdakis, S., Papadimitratos, P., & Giannetsos, A. (2015). Data Verification and Privacy-respecting User Remuneration in Mobile Crowd Sensing.

32. Lea, R. and Blackstock, M., 2014, September. Smart cities: An iot-centric approach. In Proceedings of the 2014 international workshop on web intelligence and smart sensing (pp. 1-2). ACM.

33. Correia, L.M. (2011). Smart Cities Applications and Requirements, Retrieved on October 27, 2016, from http://grow.tecnico.ulisboa.pt/wpcontent/uploads/2014/03/White_Paper_Smart_Cities_Applications.pdf

34. Chen, X., & Liu, N. (2016). Smart Parking by Mobile Crowdsensing. International Journal of Smart Home, 10(2), 219-234. https://doi.org/10.14257/ijsh.2016.10.2.21

35. Rouse, M . (2015). single-factor authentication (SFA), Retrieved on November 15, 2016, from http://searchsecurity.techtarget.com/definition/single-factorauthentication-SFA

36. Maor, E. (2014). How to Bypass Two-Factor Authentication (2FA) and What the Future Holds, Retrieved on November 15, 2016, from https://securityintelligence.com/how-to-bypass-two-factor-authentication-2fa-andwhat-the-future-holds/

37. M'Raihi, D., Machani, S., Pei, M., & Rydell, J. (2011). Totp: Time-based onetime password algorithm (No. RFC 6238).

38. Selvarajan, B. (2005). U.S. Patent Application No. 11/267,148.

39. Two-Factor Authentication Goes Mobile, Retrieved on February 27, from https://www.securenvoy.com/new-files/whitepaper.pdf

40. Margaret R., 2014 Retrieved on February, 2017 from http://searchsecurity.techtarget.com/definition/Google-Authenticator
41. Hoang, T., Choi, D., Vo, V., Nguyen, A., Nguyen, T. 2013. A lightweight gait authentication on mobile phone regardless of installation error. In: Security and Privacy Protection in Information Processing Systems, pp. 83---101. Springer, Berlin (2013). doi:10.1007/978-3-642-39218-4_7
42. Hoang, T., Choi, D., Vo, V., Nguyen, A., Nguyen, T. Gait authentication on mobile phone using biometric cryptosystem and fuzzy commitment scheme. Int. J. Inf. Secur. 14, 6 (November 2015), 549-560. DOI: http://dx.doi.org/10.1007/s10207015-0273-1
43. Shinder, D., (2001). Understanding and selecting authentication methods, Retrieved on November 17, 2016 from http://www.techrepublic.com/article/understanding-and-selecting-authenticationmethods/
44. OWASP org. (2016). Top Ten Mobile Risks, Retrieved on November 18, 2016 from https://www.owasp.org/index.php/Projects/OWASP_Mobile_Security_Project__Top_Ten_Mobile_Risks
45. Krumm J. Inference Attacks on Location Tracks. Proceedings of the International Conference on Pervasive Computing (Pervasive); Toronto, ON, Canada. 13–16 May 2007; pp. 127–143.
46. Pius Owoh, N.; Mahinderjit Singh, M.; Zaaba, Z.F. Automatic Annotation of Unlabeled Data from Smartphone-Based Motion and Location Sensors. *Sensors* **2018**, *18*, 2134. https://doi.org/10.3390/s18072134
47. Pius Owoh,N; Mahinderjit Singh, M, Applying Diffie-Hellman Algorithm to Solve Key Agreement Problem in Mobile Blockchain based Sensing Applications, Int. J. Adv. Comput. Sci. Appl, 2019
48. Zulkefli Z., Mahinderjit Singh M., Malim N.H.A.H. (2015) Advanced Persistent Threat Mitigation Using Multi Level Security – Access Control Framework. In: Gervasi O. et al. (eds) Computational Science and Its Applications -- ICCSA 2015. ICCSA 2015. Lecture Notes in Computer Science, vol 9158. Springer, Cham https://doi.org/10.1007/978-3-319-21410-8_7
49. FIDO Alliances (2018) accessed ONLINE : https://fidoalliance.org/what-is-fido/, accessed on 12/08/2019
50. Murphy C Direct assertion of identity based on a physical presence factor, 2018. Online : https://www.thedigitaltransformationpeople.com/channels/cyber-security/direct-assertion-of-identity-based-on-a-physical-presence-factor/. Accessed on : 2/03/2019