Indian Journal of Science and Technology, Vol 9(33), DOI: 10.17485/ijst/2016/v9i33/99608, August 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Disabling Camera Features in a Mobile Phone in Restricted Zones

Dhivahar Perumal*, Sharmila Sankar and M. Sandhya

B. S. Abdur Rahman University, Seethakathi Estate, Vandalur, Chennai - 600048, Tamil Nadu, India;
dhivaharperumal@gmail.com, sharmilasankar@bsau.ac.in, sandhya@bsau.ac.in

Abstract

Objective: To provide a system that is effective in its objective of curtailing the vulnerable camera component access inside the restricted environment. **Method:** Failure to provide a standardized feature for industry purpose have left them vulnerable to some of the unwanted features in a mobile phone. This means that the mobile phone is able to have all components as a casual user. This maybe a source for the violation of some of the secure company's policies. It may not be a viable option to deprive the user from possessing a phone with those components. Instead, it is a feasible solution to curb the user from using it in the restricted zone under consideration. Our solution involves a system where there is a well-defined architecture that provides the methods to curtail the vulnerable components. **Findings:** The method is an add-on on top of the existing solution and is a minimal effort to attempt at providing the necessary secure features for the organization using it. This allows the companies to enforce more productive ideas such as Bring Your Own Device (BYOD) without any concerns. **Application:** The proposed system effectively adds up to the mobile device management principles and serves the purpose of completely automated security for mobile device.

Keywords: Bring Your Own Device (BYOD), Camera Access, Mobile Device Management (MDM), Mobile Application

1. Introduction

This paper is aimed at proposing a system for a controlled access of mobile phone within a work environment. There are policies which restrict the usage of many features in a normal smart phone inside a particular zone that are otherwise considered as an unrestricted part of the everyday activity. The logical solution will be to provide a mobile phone with completely disabled or absent of features for the components that are considered vulnerable. This may be the foolproof solution under such circumstances. But, this is not always recommended given the versatility in the market and its lack of intent to create such phones with minimum scope. Therefore, our proposed solution of controlled access to the feature can be a standard for the entities to use their current device without any hassle and being approved by the organization. The organization in turn can be sure of the integrity of the policy that rules the mobile phones within its border of restriction.

The challenges to this system lies in the multiple operating systems and device technologies that are available. This means that there will be a need to provide a compatible software across all the platforms of functionality. Luckily, the commercial mobile phones fall under a very few of the operating systems. According a data from the International Data Corporation (IDC), for the quarter of 2015, 78% of the mobile phones are running under android and 18.3% and 2.7% of the mobile phones are operating in iOS and Windows respectively [1]. That makes a combined value of 99% for the three major market presence of operating system. Therefore, covering those platforms will be enough to ensure a fairly successful enough commercially thriving system.

2. Establishing the Need for the System

The purpose of the project is explained with reference to

*Author for correspondence

a few practices and policies at industry level. Explaining the need of the project, needs an insight into the overview of those policies and practices.

2.1 IT Consumerization

The usage of consumer market products of Information Technology for the business and organizational purpose is referred to as the IT Consumerization. It basically refers to the usage of technologies like mobile phones, tablets and not limited to online services like online data services, web based email services and even social networking websites. This is driven by the employees who find it favorable to link their job and work activities with their convenient technology by acquiring or in the absence of a permission from the organization. Such initiatives by the entities to contribute to the organization are welcomed as long as they don't make the organization vulnerable. It is also found that such work ethics allow the entities of an organization to contribute at improved productivity and increased business agility. In brief, IT consumerization is brought by the end users who have taken the initiative to apply the technology by their own will, outside of the influence of the IT organization and finding ways to connect with the organization with or without their approval.

2.2 Bring Your Own Device (BYOD)

The challenge with IT consumerization is that the end users tend to use their own convenient technology and this may sometime serve as a threat to the organization compromising certain security policies. The companies came up with a solution to allow the employees to bring their own devices to the work area and thus the Bring Your Own Device phenomenon allowed the company to have a standardized approach. This phenomenon gained massive acceptance due to the ever growing power of the smart phones and their ability to perform the tasks with swiftness and ease. Some of the benefits include costs saving, employee satisfaction, increased productivity and innovation. Using their own device at the workplace makes the user more comfortable with his tool and provides him an opportunity to customize according to his needs. This also means that the employee has less concerns towards working from elsewhere too. BYOD is not limited to enterprise and industrial organizations. They are being encouraged to be used in educational institutions and for other training purposes. This allows the user to highly be in touch and communicate with the peers and other faculties.

2.3 Mobile Device Management

Mobile Device Management (MDM) refers to the practice by the IT department of an organization to monitor, manage and secure the mobile devices of the employees. It refers to the range of services and products that the company uses to deploy and continuously support the various tools and applications that the employee needed in their mobile smart phones and tablets. They enforce the required policies and the level of control across the multiple platforms. Often, they provide end to end security i.e. they provide a single product that takes care of all the management from the IT side. The aim of the Mobile Device Management is to provide a secure environment to the employees by administering the required security in the devices brought in by the employees. MDM allows the administrators to oversee the mobile device like the normal desktop computers. By controlling and protecting the mobile devices, there is a reduced security risk to the company. MDM allows the employees to use the company's internal networks using a device of their choice, whilst these devices are managed remotely with minimal disruption of the other functionalities of the mobile phone. Usually the MDM software's are built with the idea to operate across multiple platforms. But sometimes there are device specific software's too which are designed based on the specific needs.

2.4 Mobile Application Management

Mobile Application Management (MAM) refers to where software's and other services are used to manage the mobile applications used by the end user in enterprise settings. MAM is typically designed over the company owned devices as well as the BYOD devices. MAM solutions typically offer a variety of capabilities and services, including application delivery and software licensing, application configuration, app authorization, usage tracking and application lifecycle management (ALM). While the MDM policies concerns itself with the securing of the device use within an enterprise, the MAM involves secures the sensitive enterprise apps and their data. MAM solution manage the device at firmware level and configuration settings and can include management of all applications and the data residing on the phone.

2.5 Challenges and Need for a Secure System

Even though it can be argued that the introduction of such innovative policies can be improvements over the non-implemented meticulous policies, there has always been a threat to the integrity of the organization. Due to this, the companies took a decline in the implementation of the BYOD policies [2]. The mobile security features are considered to be an unwanted investment from the company side when the method or system fails to provide a desired result. Since it is the employees in most cases who are bearing the cost of the devices in a BYOD environment, it is unethical to constrain their personal use of the device [3]. One such breach in the system that compromises the integrity of the companies is the highly evolving camera feature in a smartphone. The visual components possess a very serious factor of data being elicited. Today's modern algorithm can perform precision functions to gain information from the very few data available to them. This can be mitigated if we provide a secure environment where the video components does not possess the threat to the company and in turn allow the employees to continue the existing innovative policies. A result of an online survey conducted by Comp TIA in 2015 argues that 53% of the IT professionals in various sector of private businesses were not allowed BYOD compared to 34% in 2013 [4]. This may not be the death of BYOD, but carefully analyzing the facts lead to the conclusion that the company is more concerned of the threats that are coming along with those policies rather than the perks when they are implemented [5]. It is also to be mentioned that digital video technologies in the smart phone is not the desired feature in any organization environment, with most of the functions involve graphic contents at the most [6]. But it is hardly possible to find a mobile phone in the market that comes without a camera component. Thus, the solution must be a foolproof methodology to avoid the data leakage rather than completely abandon the BYOD policies.

3. Proposed Solution

As displayed in Figure1 the system has an application which is designed in their respective operating system by using the respective software development toolkits. There are Application Program Interfaces that are designed to be used by the developers to develop an application that can run on the mobile phone. The implementation of the application may be operating system specific, but the underlying concept is the same and is perfectly possible to provide a fault tolerant system.

The application which resides in the mobile phone user side is not of any functional significance to the user. Several protective steps had to be taken for preventing the user to bypass the security as well from nullifying the application. The application must comply with the various restrictions provided by the operating system itself. This is achieved by using a mechanism that constantly monitors the parameters associated with the phone and sends a distress message to the server if any violation is attempted. The application serves also as a listener to the server which controls the access of certain components when the device is inside the restricted zone. The application is cryptic to the access and can be controlled by the server alone.

The server resides in the restricted zone and monitors the incoming and outgoing of the mobile devices. The server holds the data about all the registered mobile phones. The server on authentication by the mobile phone sends the respective commands that invokes camera access denying code. This denial of camera access is maintained as long as another command to revoke its status is sent from the server. The server identifies the mobile phone by its unique IMEI number. Thus the detailed information about the phone's activity can be tracked by the server.

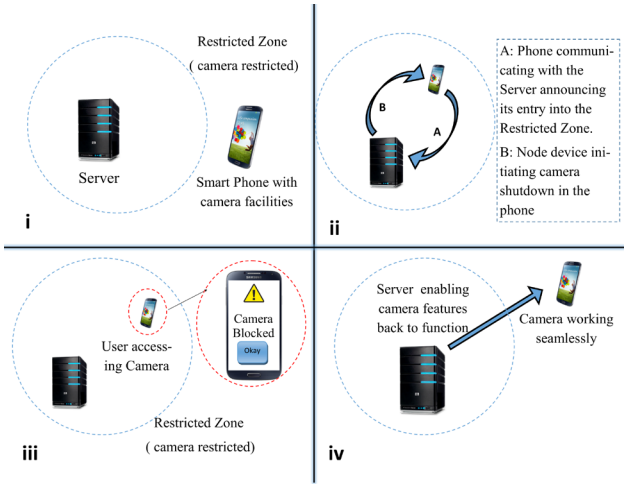

Figure 1. The proposed solution to block the camera access. i) The setup of the server in the restricted zone and the mobile phone equipped with the application. ii) The communication between server and the application which shuts down the camera access. iii) Camera block status displayed when the user tries to access the camera inside the zone. iv) Server enabling the mobile phone to access the camera.

4. Product Design

The system consists of two ends of controls which makes the managing of the camera access in the smartphone possible. The two ends are designed cohesively so that they can be easily made to be functional for different platforms.

4.1 Mobile Application

The first control resides in the mobile phone of the client and is in the form a native application. This application has a higher control to other ordinary application. The application has device admin control. This enables the mobile phone to have a control over the smart phones multiple hostile features like camera access and password policies. In the case of android, this is achieved by importing the device administration API after getting hold of the permission to implement it in the android manifest file. This application is installed pre in hand when the smart phone is to be registered in the network. The mobile phone is identified by its unique IMEI number. Every phone that has to enter the zone has to be registered compulsorily. The unique identification number serves as the ID that keeps the other information being collected associated with it in the server.

When the smart phone enters the zone, there is a trigger in the application that is activated at the entry. The trigger runs the code at the background that disables the camera features of the phone. This is done by the logics available in the respective platform. In android, once we get hold of the device admin permissions, we are presented with the classes that allows us to build the application with camera control. The respective camera access code is called and the smart phone is kept hold of the blocking status till the phone is set to leave the zone. Between the time period, if the user tries to access the phone either from the native camera application or from any application that uses the camera feature, it renders an error message. Now this is a high level locking feature provided by the operating system itself. But in some cases, in the absence of such native codes to develop the application, we can improvise another method which helps to solve the situation. The current application acquires the camera access and locks its resource. Now another application can get hold of the camera access only after the current application releases it. As long as the current application holds the access, the other application cannot even try to get hold of it. This can be released by the application when the user gets out of the zone.

The application must be receptive to attempts that leads to uninstalling or bypassing the application. This is ensured by using a monitor that tracks of the mobile phone usage. Any attempt to uninstall the application inside the will make a distress message to the server temporarily locking down the device's camera feature. The server tags the IMEI code of the relevant phone with the activities attempted. When entering the zone, when the application is absent, the server doesn't acknowledge the smart phone and forces to install the application. Thus once the application is installed, there are few chances to uninstall the application. If the user is able to bypass all the mentioned security levels and able to access the camera, then the application tracks the camera usage and attempts to listen to every information captured and sends a detailed log of the data to the server. This is done by listening to the intents that involve camera capture and in turn handling them to send the required information.

The camera access is retained when the mobile phone exits the restricted zone. The mobile phone receives a command code which is triggered from the server. The mobile application calls back the device admin command or releases the camera access according to the method implemented initially. In addition to that, any log of data stored when the camera was accessed inside the restricted zone despite the security lock is sent to the server.

4.2 Server Software

The other end of the control resides in the server system at the restricted zone. This system handles all the information in and out of the zone. It has the database of mobile phones registered in the network. The mobile phones are uniquely identified by their IMEI number. The server software listens for the entry of the mobile smart phones entering the network through the entry mechanism which is to be discussed later. The mobile phones which enters the zone sends its IMEI number to match with the database. A string of data relating to the security implementations are also received by the server. On acknowledgement from the data received, the server sends the command to lock the camera. The mobile application returns the acknowledgment that the camera feature is blocked successfully. The server checks for the attempts to bypass or uninstall the application. The server receives the data from the mobile application per-

taining to the vulnerable actions and then responds to it. On uninstalling the mobile application, the server flags the corresponding smart phone entry in the database. The other threat of bypassing the application is handled by receiving the information from the mobile application regarding the capture of images by the camera while the phone is inside the restricted zone. This is stored in the application log and received when the smart phones logs out from the restricted zone. If a situation arises when the user has uninstalled the application inside the restricted zone and tried to access the camera and install it back before quitting the zone. All these data are stored in a remote log too. These maybe received by the server on logging out as well.

4.3 Connectivity

The next challenge is to provide a seamless connectivity between the server and the mobile phone. Using a mobile data network for internet connectivity is not a viable and logical solution. Instead, we can provide our own WiFi Access point and a captive portal through which we can access the mobile application at least at the point of entry and exit. As long as there is an internet connectivity there can be a trigger invoked from the server on the mobile application. The future of this product design deals with continuous connectivity. But currently the product is designed to function when there is a log in and log out from the end user alone. There are Wi-Fi Access points set up using minimalized setup hosting a captive portal. The portal points out to a web page that allows to download the application and register for the first time. The registered user information residing in the server is then verified and then a camera block is initiated. After that there is no need for the internet to keep track of the application activities. The application itself is designed to provide the necessary mechanism to prevent the user from malpractices. On log out the user has to again connect to the Wi-Fi access point provided and request the server. The server acknowledges the mobile phone and releases its hold. On either time, if there are additional log of data in the application, they are being sent to the server.

5. Systematic Working

The various point of design is seamlessly coupled to work in a functional environment. It is in the functional protocol that the actual secureness of the design relies upon. The system has to consider two types of entities. One set of people are first time entries to the restricted zone, while the others are pre-registered to the server.

5.1 Registering to the Server

The first time users are to register to the network for gaining access to the restricted zone. Since our server connects to the mobile phone through internet connectivity, we need to provide the user with the internet connectivity through the custom WiFi Access Point. This is because the link to the application is residing on the captive portal on the particular resource. The user is able to download the application and install it. On check-in the server automatically adds for a new entry if it doesn't already exist in its database.

5.2 Entry into the Restricted Zone

The end user opens up the application and select the option for entering into the zone. This connects the server for matching its entry with respect to its IMEI number. The server acknowledges and sends a command back to the application to lock the camera access. In addition to that, there is a unique code sent to the application that releases the camera access only when the same code is sent over again to the application. The code is handled as hash which render any bypassed information irrelevant.

5.3 Inside the Zone

The camera access of the mobile is completely shut off inside the restricted zone. The camera access is not only held in the native applications but also in any application that requests for it. Any attempt to bypass the security will be noticed on the way out. The system catches all the logs created while exiting out of the application. If the application was tried to be compromised by removing it, it will have caught in its log data.

5.4 Exiting the Zone

The application requests the server to release the camera block on the device on about to exit the zone. When the confirmation is given by the application, the application in the smart phone releases the control back. Thus the camera access in the phone appears to function normally. The smart phone is not found to have any problem in accessing its camera from the native camera application as well as any other application that uses camera facility.

6. Security Approaches

The system has to withstand certain obvious concerns that can put its functioning in jeopardy [7]. These scenarios are carefully analyzed and certain improvements have been made over the system. This vouches for the integrity of the system to be used commercially. The typical scenarios where an end user can attempt to compromise the security are as follows.

6.1 Scenario 1: User Tries to Uninstall Inside Restricted Zone

In this scenario, the user tries to bypass the security hold by uninstalling the application inside the restricted zone after entering into it. The application cannot be uninstalled due to the device admin rights to it. If the end user is successful in terminating, then he might be notified when he tries to exit the zone. Moreover, the application tries to lock down all its access and sends a distress message to the server to notify the system of its breach.

6.2 Scenario 2: User Tries to Uninstall Inside Restricted Zone and Reinstalls before Exiting

The user may reinstall the application from the downloaded file earlier used to install it. The user maybe having the application in his smartphone during the exit, but there will be log created during the file and it may notify the server when the user tries to exit at the zone. The log has detailed information stored whenever it is tried to be stopped.

6.3 Scenario 2: User tries to Bypass the Security Code to Unlock the Camera Access

The user may listen to the data sent from the server initially and then may try to use it to unlock the device. But this is not possible because of the authentication being carried out by a cryptographic hash function. This renders the eve droppers no significant information to unlock the camera access later.

7. Conclusion

The system is fully capable of acting as a security solution with just the authentications provided at the entry and exit. This serves as a standalone solution to the industries who are concerned with the BYOD policy. The system provides an intact solution and satisfies all aspects of the security approaches. But the failure to provide a common standard of approach for different platforms can be solved by introducing a standard specific to the current need. The system can be strengthened and a much simpler methodology is possible if they do so.

8. References

1. Smartphone OS Market Share [Internet]. [Cited 2015 Sep 13]. Available from: http://www.idc.com/prodserv/smartphone-os-market-share.jsp.
2. The bring-your-own-device fad is fading [Internet]. [Cited 2013 Oct 10]. Available from: http://www.computerworld.com/article/2948470/byod/the-bring-your-own-device-fad-is-fading.html.
3. Miller KW, Voas J, Hurlburt GF. BYOD: Security and privacy considerations. IT Professional. 2012 Sep–Oct; 14(5):53–5.
4. Guoyou H. Requirements for security in home environments. Residential and Virtual Home Environments Seminar on Internetworking. Spring; 2002. p. 1–17.
5. Kulkarni G, Shelke R, Palwe R, Solanke V, Belsare S, Mohite S. Mobile cloud computing-bring your own device. Communication Systems and Network Technologies (CSNT), Fourth International Conference; 2014. p. 565–8.
6. Gessner D, Girao J, Karame G, Li W. Towards a user-friendly security-enhancing BYOD solution. NEC Technical Journal. 2013; 7(3):113–16.
7. Richard O. Why the BYOD boom is changing how we think about business it. Engineering and Technology. 2012; 7(10):1–28.