Disabling Camera Features in Mobile Phone in Restricted Zones

Mr. Omkar Gaonkar[1,a] , Prof. Sujata Patil[2]

[1]Dept of MCA-Trinity Academy of Engineering, Pune, India

[2]Assitant Professor, Trinity Academy of Engineering, Pune, India

[a]Omkargaonkar920@gmail.com

Abstract

An application to provide a system that is effective in its objective of reducing the vulnerable camera component access inside the restricted environment.

The proposed system effectively adds up to the mobile device management principles and serves the purpose of completely automated security for mobile device.

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.

Objective

1. **Heighten Security:** The principal aim of implementing mobile camera blocking in restricted areas is to reinforce security measures within these zones, thwarting unauthorized photography or recording of confidential information.
2. **Preserve Privacy:** By disabling mobile camera functionality, the objective is to safeguard individuals' privacy within restricted zones, ensuring that confidential tasks, conversations, or data remain undisclosed and shielded from potential breaches.
3. **Mitigate Information Leakage:** Mobile camera blocking serves to prevent both accidental and deliberate capture of sensitive data, proprietary information, or classified materials, thus lowering the risk of information leakage or corporate espionage.
4. **Ensure Compliance:** Enforcing camera blocking aligns with regulatory standards and organizational policies governing data protection and confidentiality, thereby mitigating legal risks and upholding adherence to industry regulations.

5. **Protect Intellectual Property:** Restricting mobile camera access in restricted areas safeguards intellectual property, trade secrets, research findings, and other proprietary assets from unauthorized replication, distribution, or exposure.

Methodology

we can see the system comprising an application tailored for specific operating systems using corresponding software development toolkits. This application provides APIs for developers to create mobile phone applications. While the implementation may vary per operating system, the fundamental concept remains consistent, ensuring a fault-tolerant system. The mobile application, though inconspicuous to users, requires careful measures to prevent security breaches or attempts to nullify it. It must adhere to operating system restrictions, achieved through continuous monitoring of phone parameters. Any violation prompts the application to send a distress signal to the server. Moreover, the application acts as a liaison with the server, regulating access to specific features when the device enters restricted zones. Only the server can control the application, which remains cryptic to user access. Positioned within the restricted zone, the server tracks incoming and outgoing mobile device activity based on their unique IMEI numbers. Upon authentication, the server issues commands to deny camera access, persisting until instructed otherwise. Consequently, the server retains comprehensive data on the phone's activities.

Implementation

1. Mobile Application
   The primary control mechanism is embedded within the client's mobile phone as a native application, endowed with elevated privileges compared to regular apps. This application functions as a device administrator, granting the mobile device authority over various potentially harmful features such as camera access and password protocols. For instance, in the case of Android devices, this is achieved by integrating the device administration API into the Android manifest file post obtaining requisite permissions. The native application is pre-installed prior to the smartphone's registration on the network, with each device uniquely identified by its IMEI number. Registration is mandatory for every device entering the designated zone, facilitating association of collected data with its corresponding IMEI on the server. Upon entry into the zone, the application activates a trigger, initiating background processes to disable the phone's camera functionality using platform-specific logic. Access to camera controls is facilitated through the device admin permissions, allowing the application to enforce camera restrictions until the device exits the restricted area. Attempts to access the camera during this period, whether through the native camera app or other applications, result in error messages, courtesy of the operating system's built-in security measures. In instances where native code is unavailable, alternative methods are improvised, wherein the application monopolizes camera access until released, preventing other apps from accessing it. The application remains vigilant against uninstallation or circumvention attempts, promptly notifying the server of any such activities to temporarily lock down camera access. In the absence of the application within the restricted zone, the server refuses acknowledgment to the smartphone, compelling installation of the requisite application. Once installed, the application's robust security measures make uninstallation unlikely. Any successful attempts to bypass security measures and access the camera trigger the application to log the activity and relay detailed information to the server. This is achieved by intercepting camera capture intents and transmitting pertinent data accordingly. Camera access is restored upon the device exiting the restricted zone, triggered by a command from the server, with the application reverting camera permissions as per the initially implemented method. Furthermore, any data logs generated during unauthorized camera usage within the restricted zone are promptly forwarded to the server.

2. Server Software

   On the opposite end of the spectrum, the control mechanism is situated within the server system located in the restricted zone. This system is responsible for managing all inbound and outbound data within the zone, including the database containing registered mobile phones. Each mobile phone is uniquely identified by its IMEI number. The server software actively monitors the entry of mobile smartphones into the network via a mechanism to be discussed later. Upon entering the zone, mobile phones transmit their IMEI numbers to be matched with the database. Additionally, the server receives a set of data concerning security implementations. Upon confirming the received data, the server issues commands to lock the camera feature. The mobile application acknowledges the successful blocking of the camera feature. The server also scrutinizes attempts to bypass or uninstall the application, responding accordingly based on data received from the mobile application. In the event of application uninstallation, the server marks the corresponding smartphone entry in the database. Furthermore, to address the threat of application bypassing, the server relies on information provided by the mobile application regarding any camera captures within the restricted zone, which is logged and retrieved upon the smartphone's logout from the zone. In scenarios where a user uninstalls the application within the restricted zone and attempts to reinstall it before exiting the zone, such data are stored in a remote log accessible to the server upon logout as well.

3. Connectivity

   The upcoming challenge involves ensuring smooth connectivity between the server and the mobile phone. Relying solely on a mobile data network for internet access isn't practical. Instead, we propose establishing our own Wi-Fi access point along with a captive portal, facilitating access to the mobile application at entry and exit points. Whenever there's internet connectivity, the server can trigger actions on the mobile application. While the product's future entails continuous connectivity, its current design operates upon user login and logout events exclusively. We've set up Wi-Fi access points with simplified configurations, hosting captive portals directing users to a webpage for application download and initial registration. Once user details are verified against server records, camera blocking is activated. After this setup phase, internet connectivity isn't necessary to monitor application activities, as the application itself incorporates mechanisms to deter misuse. Upon logging out, users must reconnect to the provided Wi-Fi access point and request server acknowledgment to release their device. During either login or logout, any additional application data logs are transmitted to the server for storage and analysis.

Conclusion

The system offers comprehensive security measures solely relying on the authentication processes at entry and exit points. This autonomy renders it an ideal standalone security solution, particularly beneficial for industries implementing BYOD policies. It effectively addresses various security concerns and aligns with diverse security approaches. However, addressing the lack of a universal standard for different platforms could enhance the system's effectiveness. Introducing a standard tailored to current requirements could bolster the system's capabilities and streamline its methodology significantly.

References

www.quora.com
www.researchgate.net
https://patents.google.com/patent/US8219144B2/en
https://www.42gears.com/solutions/capabilities/intelligent-camera-blocking/