Open Access Library Journal
2022, Volume 9, e9264
ISSN Online: 2333-9721
ISSN Print: 2333-9705

Secure Cloud Based Approach for Mobile Devices User Data

Oscar Mbae, David Mwathi, Edna Too

Computer Science, Chuka University, Chuka, Kenya
Email: karugaoscar@gmail.com

How to cite this paper: Mbae, O., Mwathi, D. and Too, E. (2022) Secure Cloud Based Approach for Mobile Devices User Data. *Open Access Library Journal*, **9**: e9264.
https://doi.org/10.4236/oalib.1109264

Received: August 29, 2022
Accepted: September 24, 2022
Published: September 27, 2022

Abstract

In this era characterized by rapid technological innovations, mobile devices such as tablets and smartphones have become inevitable due to the variety of services they offer. These devices are susceptible to loss due to their small portable sizes. The personal identification numbers (PIN) and patterns used as security controls have minimal encryption. One of the main challenges of using mobile devices is the risk of private and/or confidential data stored in the device's internal memory being exposed to unauthorized persons and the risk of permanent loss and/or damage of that data. The objective of the study is to analyze existing security solutions for mobile devices user data storage and propose a secure approach to address possible security threats and vulnerabilities to these solutions. The solution provides mobile device users with the ability to securely store/retrieve their data anytime and from anywhere even in the event of device loss. Furthermore, mobile device manufacturers are able to leverage on the solution and patch their systems in order to improve on mobile device data storage security. The research highlights existing threats and vulnerabilities to existing mobile data storage devices. This helps in creating awareness to mobile device users who as a result are expected to be more vigilant and can also utilize proposed precautionary measures.

Subject Areas

Securing Information Technology, Cloud and Services Computing, Connection of Everyday Objects to the Internet

Keywords

Threats, Vulnerabilities, Mobile Data Storage, Data Security

1. Introduction

Information systems are an integrated set of components for collecting, storing,

and processing data in so as to afford knowledge, information, and digital products (Pearlson *et al.*, 2019) [1]. Most people and organizations rely heavily on information systems to manage, compete, interact as well as carry out their operations on a daily basis. Governments set out these information systems to provide services cost-effectively to their citizens. Digital products and services such as social networking, electronic books, software, video products and gaming are delivered by information systems. There are different types of devices used by information systems such as; mobile phones, desktop computers, laptops and tablet computers (Haynes, 2015) [2]. Notably, mobile devices are the generally utilized because of their versatility (Haynes, 2015) [2].

With the growing use of mobile devices and internet, there has been rapid increase in the number of application that has been developed to meet the needs and demands of the users (Ramalingam *et al.*, 2017) [3]. These applications have been found to gain access and disclosed sensitive information from the users' devices by accessing them through the android permissions granted (Aldossary & Allen, 2016) [4]. Despite security measures ensuring that malicious applications do not make it to the store, disguised applications still make it through the security screening process to the users' mobile devices. A research conducted on privacy and data management on mobile devices, indicates that there has been an increase in the malicious mobile applications and other repackaged applications acting on behalf of the adversary's that execute on them (Yalew, 2018) [5]. Therefore, with easily accessible file systems, a malware can get to examine the user's confidential data and steal it, hence, compromising the security of the clients.

Due to the ubiquitous nature of the mobile device and the wealth of the applications that are available for use, smartphones have become a reliable indicator of our everyday lives (Mollah *et al.*, 2017) [6]. The devices, however, do not guarantee the confidentiality of the information that is contained in its storage. It is the users' responsibility to ensure that confidentiality is achieved. Therefore, the security risks/threats and vulnerabilities that are associated with the mobile devices strip the users the right to privacy.

In the bid to secure data on smartphones, developers have come up with application with lock patterns, pins and use of finger print readers to prevent unauthorized access to it (Gupta *et al.*, 2018) [7]. However, further research established that patterns and pins can easily be cracked by hackers to grant them access to the data in smartphones (Padma & Kumar, 2016) [8]. There are still aspects of security concerns that have come up such as, loss of a device and theft. Moreover, with internet connection, these devices are at risk of hackers accessing and reading the data contained in them. Additionally, the loss of data owing to hardware failure as well as users being locked out of the mobile device has been quite rampant recently (Murugesan, 2019) [9], with the emerging way being the ransom ware attack.

Sensitive and confidential data stored in an unprotected device can be ac-

cessed easily without authorization resulting to the information leakage. (Franchi *et al.*, 2015) [10]. Hence, the general objective of this paper is, to analyze the potential data security threats and vulnerabilities associated with mobile data storage.

2. Previous/Related Work

Encryption as an IT strategy for achieving confidentiality and availability of data, is among the viable solutions used to protect the data that lives in and between the devices. Encryption is considered as a very powerful and reliable way of keeping the user's data secure, and while it is penetrable, it's a great deterrent to hackers (Kumari, 2017) [11]. In the event data is stolen, it becomes unreadable and almost worthless if it is encrypted. Mobile device manufacturers together with the operating systems developers have incorporated encryptions on the devices to be utilized by the users (Heath, 2018) [12]. The available encryptions solutions can be achieved through; Full device encryption (FDE): Pre-encrypting data and file encryption is the primary way to protect the mobile devices and the at-rest data stored there, where any files kept in the device are automatically encrypted. Pre-encrypting data that is synchronized with the cloud: there is a great deal of software available that can get to pre-encrypt data before its synchronized to the cloud, making it obscure to the cloud or any other person that tries to access it. This leaves the files available on the device unencrypted and is still vulnerable. File encryption provides a means to encrypt data at-rest on a file by file basis so that it cannot be read in case it is intercepted. This is not automatic, however, it is beneficial as that data remains encrypted after leaving its origin.

As a common solution to users on the availability and confidentiality of data, different mobile manufacturers have come up with different mobile solutions for this purpose. Existing solutions include: Safe for Huawei p9 and Later Models, Samsung Android Mobile Device Security, Google cloud storage, Microsoft OneDrive, Signal, CoverMe, Secret Box and Crypt4All.

2.1. Safe for Huawei p9 and Later Models

This is a secure mobile storage folder that is inbuilt in the Huawei P9 models devices. It stores data of type image, Audio, videos and files in pdf formats.

Huawei's HiSilicon chips provide hardware-based, chip-level security protection. They avert side-channel attacks as well as other physical attacks via a range of security capabilities: secure storage, secure boot, the True Random Number Generator, hardware-based algorithmic engines, a trusted execution environment and hardware-level attack prevention (Busch *et al.*, 2020) [13]. Additionally, these chips operate in a trusted execution environment so as to protect data, device systems and network communications. Huawei's innovative financial-grade inSE security solution embeds a security chip into a smartphone processor. inSE utilizes software algorithms and a System-on-a-Chip (SOC) design to protect both hardware and software (Ning & Zhang, 2019) [14]. This ensures

chip-grade protection for a smartphone's system security and user privacy because it can build defenses into software and withstand hardware attacks.

The safe solutions function in a way that the users creates a key that would be used to self-encrypt the files selected (L. Zhang & Chen, 2018) [15]. The phone finger prints are also linked to the storage for use to unlock it. In this case, the key is usually stored on the device, and this makes it vulnerable for those devices that have been rooted (Park *et al.*, 2019) [16]. Additionally, the safe does not provide self-destruction capabilities and ease of accessibility in the event the device is lost or misplaced. Therefore, in the event one misplaces the device or forgets to close a file after viewing it unauthorized persons can easily gain access.

2.2. Samsung Android Mobile Device Security

Users usually just need to set up an authentication method, through either Password, Pattern, Pin, or Biometric based on the model of the phone (Ramalingam *et al.*, 2017) [3]. Once the user is authenticated, granted access enabling installation of applications and data to be created directly into a Secure Folder, or relocated to Secure Folder from outside the container to be secured (Kanonov & Wool, 2016) [17]. The Samsung file encryption allows for encryption of data stored in both the mobile device and the memory card. It uses the AES 256-bit cypher algorithm to accomplish its objective (Lu *et al.*, 2016) [18]. The Samsung secure folder is quite user friendly, although it is limiting to the phone model, to be able to decrypt and access the data. There is inability to upgrade the previous models of the devices, to support the application, this is thus limiting to few users. In case of device loss, the data is still contained in the local memory, although in unreadable format, and it would take a lot of persistence in cracking up the algorithm in use to be able to access the data, which may take a while. Therefore, making its accessibility/availability limited to one device. However, the solution does not provide for self-destruction abilities after the user finishes accessing the data. As a result, if the user forgets to close or delete viewed content from history, one can easily get unauthorized access to it.

2.3. Google Cloud Storage

Google cloud storage is a web-based file storage platform used for storing and accessing user data on Google cloud infrastructure. It as well provides a variety of associated services such as sharing the online saved data and downloading it to your personal computer. Google cloud storage utilizes AES algorithm to encrypt the data at-rest. Data at the storage level is encrypted using AES256 by default, except for a small number of persistent disks created earlier than 2015 that use AES128. Google cloud initial security level involves authentication of login username and password (client side encryption), as well as, two-step verification whereby a code is sent to the users' phone which they must enter before being given access or a tap request that the user taps to accept access (Henziger & Carlsson, 2019) [19].

Secondly, Google cloud storage gets to encrypt data uploaded from your personal computer when storing it online (server side encryption). Data stored in Google Cloud Platform is normally separated into sub-files for better and safer storage, each chunk is encrypted at the storage level with an individual encryption key (Jayapandian & Md Zubair Rahman, 2018) [20]. The key used to encrypt the data in a chunk is referred to as a data encryption key (DEK). However, due to the high capacity of keys at Google, and the necessity for high availability and low latency, these keys are stored nearer to the data that they encrypt. The DEKs are encrypted with a key encryption key (KEK). Clients choose which key management solution they prefer for managing the KEKs that protect the DEKs that protect their data (Jayapandian *et al.*, 2016) [21]. This encryption process makes it difficult for hackers who gain access into their servers from viewing client information.

However, due to the many services it provides, it is vulnerable to Google cloud data breach. This is caused by human error whereby a user wrongly sets the sharing settings for a file as public and ends up sharing private data publicly or email the wrong file to the wrong person (Chu *et al.*, 2013) [22]. Secondly, despite server side encryption protecting client data from hackers, once login details are verified, any other party can easily access the cloud-based information. The addition encryption steps are optional for the user and as a result some do not utilize them to protect their confidential information. Notably, the solution does not provide self-destruction capabilities, therefore, it is usually up to the mobile device user to delete all data files after downloading and viewing them.

2.4. Microsoft OneDrive

Microsoft one drive is part of a web version of office operated by Microsoft that provides file hosting and synchronization services (Gelb, 2014) [23]. Microsoft offers service side technologies that are used to encrypt customer data in transit and at rest. Microsoft OneDrive Encryption for at rest data comprises of two components: Per-file encryption of client content and BitLocker disk-level encryption (Lonsky, 2018) [24]. BitLocker is deployed for OneDrive for Business as well as SharePoint Online across the service. In addition, per-file encryption is also in SharePoint Online in Microsoft 365 multi-tenant and OneDrive for Business and new dedicated environments built on multi-tenant technology (Ferdaus *et al.*, n.d.) [25].

Whereas BitLocker encrypts all of the data on a disk, per-file encryption goes an extra step where it includes a distinctive encryption key for individual file. Furthermore, each update to every file is given its own encryption key. Before storage, the keys to the encrypted content are then saved in a different physical location from the content. All the steps of this encryption use AES 256 and are Federal Information Processing Standard (FIPS) 140-2 compliant. The encrypted content is normally dispersed across several containers throughout the datacenter, and each container has its unique credentials (Lonsky, 2018) [24].

These credentials are as well stored in a physically separate location from either the content or the content keys.

Microsoft one drive provides a number of optional encryption and account verification steps that users can use to protect their data (Henziger & Carlsson, 2019) [19]. These steps include;

- Creating of a strong password
- Adding security information to your OneDrive account such as phone number, email address and security question and answer. This information helps a user to access their account in the event it is hacked into and password changed or the user forgets login information.
- Two factor verification, this technique when chosen requires the user to enter a security code whenever they sign in from an untrusted device.
- Enabling encryption on your mobile device, this involves use of fingerprint, pins and lock patterns for users using OneDrive mobile application to enhance the security on their devices.

Just like Google cloud, due to the many services it provides it is vulnerable to OneDrive data breach. Secondly, despite server side encryption protecting client data from hackers, once client communication with the server process verifies login details, any other party can easily access the cloud-based information. The addition encryption steps are optional for use as well, as a result, some do not use them to protect their confidential information. Additionally, the solution does not provide self-destruction capabilities, therefore, it is usually up to the mobile device user to delete all data files after downloading and viewing them.

2.5. Signal

Signal is an end-to-end encrypted voice calling encryption application used by iPhone (Ermoshina *et al.*, 2016) [26]. It was designed and created by the open source encryption software group known as Open Whisper Systems. Additionally, the application also offers end-to-end encrypted text messaging. This provides for a calling and texting experience that is worry free, secure and private (Ermoshina *et al.*, 2016) [26]. As a result, no one else except you and your recipient can decipher your words.

The underlying crypto system on which Signal is built is known as the Signal protocol. The Signal Protocol amalgamates the Double Ratchet algorithm, Extended Triple Diffie-Hellman (X3DH) key agreement protocol, pre-keys, and uses AES-256, Curve25519 and HMAC-SHA256 as cryptographic primitives (Cohn-Gordon *et al.*, 2020) [27]. It generates a long term set of identity key pair, medium-term signed pre-key pair, as well as numerous ephemeral pre-key pairs. These keys are usually generated at the client side then locally stored in a secure location. The next step includes packaging all of the registration ID and public keys into an object referred to as the ("key bundle") and registering it with a Key Distribution Center (Cohn-Gordon *et al.*, 2017) [28]. In order for Alice to send messages to Bob, Alice must know and have access to Bob's registration ID and

public keys to start a session. Thus, Alice must first generate her own keys and register herself with the key distribution center and request Bob's key bundle.

However, the application does not encrypt other forms of content such as user files and other documents in different formats (Abu-Salma *et al.*, 2017) [29]. The solution does not also offer automated self-destruction capabilities. As a result, users cannot protect the privacy of their private files stored in the mobile device from being accessed by unauthorized persons.

2.6. CoverMe

CoverMe is an application that encrypts voice messages, text messages and user files by use of passwords (X. Zhang *et al.*, 2017a) [30]. Notably, it as well provides the user with an option to manually delete text messages by remotely sending a wipe instruction to remove it. CoverMe also has an encrypted vault for data storage that ensures personal contacts, messages, call logs, confidential documents, pictures, and videos stay hidden and private (Schneier *et al.*, 2016) [31]. Additionally, in the event you are forced to share your password, a decoy password can also be set up. CoverMe uses numerous encryption algorithms such as RSA, AES 128 and AES 256 (X. Zhang *et al.*, 2017b) [30].

However, the application only gets to delete messages manually and the user data is lost in the event the mobile device is lost as the content is stored in its local memory. As a result, Accessibility/availability of data is limited as users cannot retrieve their data without access to that specific mobile device used to store it. Additionally, self-destruct capability is only available for text messages but not to user files.

2.7. Secret Box and Crypt4All

They utilize the AES 256 cipher algorithm in encryption of their data (Bursać *et al.*, n.d.) [32]. In secret box, it sets an encrypted area on the phone's hard drive or the media card, and a master password is then set that is used to open the secure area. Backup of the data is achieved by making a copy on the media copy for use, it thus should be done manually. Unlike Secret Box, whose data is stored in the device's hard drive. Crypt4all gets to store a copy of its encrypted files in an online database (Ritacco & Wills, 2018) [33]. This makes it easier for users to retrieve the files when there is need. The process of retrieval is however consuming, since the files can't be viewed directly from the solution. It should be decrypted, and then saved on the phones drive for viewing (Luangoudom *et al.*, 2019) [34]. The users should then delete the file once again after viewing. This makes it quite vulnerable, as usually deleted sensitive files can usually be recovered and viewed by attackers when they get access to the drive. The solutions discussed above get to provide a solution for confidentiality of data. However, their approaches still place the files at a vulnerable position, as the solutions get to run on the same execution environment as the malware that infects the device (Lu *et al.*, 2016) [18]. The encryption keys stored in the internal memory of the

device are also quite vulnerable to being accessed as they reside on the libraries that are on the same environment as the malware on the device.

3. Methodology

Mixed methods were employed in order to achieve the intended objectives.

The data collection and analysis procedure involved;

1) Downloading of existing mobile device user data security tools/solutions.

2) Installation of those tools on an android environment.

3) Manipulation of the tools while observing their behavior.

4) Recording presence, absence or status of security parameters under observation.

5) Determining possible threats that could potentially exploit the weaknesses of these tools.

6) Propose a secure approach for mobile devices user data.

7) Prototype the proposed solution.

8) Validate the prototype based on presence, absence or status of security parameters under observation and compare with the existing solutions.

4. Findings

Table 1 shows an analysis of Existing Security Solutions for Mobile Devices User Data Storage.

4.1. Proposed Solution

The proposed solution, as shown in Figure 1, adopts the AES 256 encryption technique among other additional encryption methods like the hash function, Password Based Key Derivation function and the Salt function. The solution concentrates on securing the data on the mobile devices by storing it in an encrypted format and uploading it to the cloud. Additionally, the solution is timed for the downloaded shadow copy to self-destruct after user consumption, eliminating unauthorized person or application from reading the information without a decryption key.

The proposed mobile devices user data security solution leverages on sampled existing models. It seeks to improve on mobile device data security by storing mobile user data online instead of on the mobile device memory, with additional encryption steps. Sensitive and confidential data stored in an unprotected device can be accessed easily without authorization resulting to the information leakage. (Franchi *et al.*, 2015) [10]. The mobile data privacy solution proposed by this research strives to provide a security solution to users to be able to store sensitive data and access it on their mobile devices.

In accomplishing the objective of offering encryption and availing the data whenever needed, the proposed mobile device user data security solution should be able to cater for the users' needs effectively. The core desirable design features of the solution are;

Table 1. Shows an analysis of existing security solutions for mobile devices user data storage.

	Solutions	Automatic File Self-Destruction capabilities	Data Storage Location	Encryption Method Employed	Strength of Encryption	Login Authentication required	Data recovery in the event of mobile device loss	Access control based on data confidentiality level required	Possible Treats/ Vulnerabilities
1	Safe for Huawei	No	Mobile device	Huawei HiSilicon chips	AVERAGE	Yes	No	NO	-Sniffing -Reverse engineering -loss of confidential data due to mobile device loss.
2	Samsung Android Mobile Device Security	No	Mobile device	AES 256	AVERAGE	Yes	No	NO	-Insufficient Authorization and Authentication Controls -Sniffing -Loss of confidential data due to mobile device loss.
3	Google cloud storage	No	Cloud	AES 256	HIGH	Yes	Yes	NO	-Insufficient Authorization and Authentication Controls -Sniffing -Remote code execution -privilege escalation
4	Microsoft OneDrive	No	Cloud	AES 256	HIGH	Yes	Yes	NO	-Sniffing -Insufficient Authorization and Authentication Controls -Remote code execution -privilege escalation

Continued

5	Signal	No	Cloud	Signal protocol	HIGH	Yes	No	NO	-Insufficient Authorization and Authentication Controls -Sniffing -Reverse Engineering
6	CoverMe	No	Mobile device	AES 128 and AES 256	AVERAGE	Yes	No	NO	-Reverse Engineering -Client Code Security -Insecure communication -Loss of confidential data due to mobile device loss. -Sniffing
7	Secret Box	No	Mobile device	AES 256	AVERAGE	Yes	No	NO	-Sniffing -Reverse Engineering -Loss of confidential data due to mobile device loss.
8	Crypt4All	NO	Mobile device and cloud	AES 256	AVERAGE	YES	YES	NO	-Reverse Engineering -Client Code Security -Loss of confidential data due to mobile device loss.

- User login authentication functionality: In this case, an account will be created by the user to allow them to login to the system. The password will be hashed for safety so that it is not stored in plain text. Additionally, the login details will be verified to ensure they match that specific account.
- Authorization/Access control: After successful login into the system, users will be required to input a decryption key in order to view online stored data.
- Accessibility and availability from anywhere: Users get to access their data anytime from anywhere even in the event their mobile device is lost. All they require is access to another mobile device and login using their account details.
- Secure data: data stored in the database is in encrypted format, so that malicious users would not be able to read it even if they hack the database. The

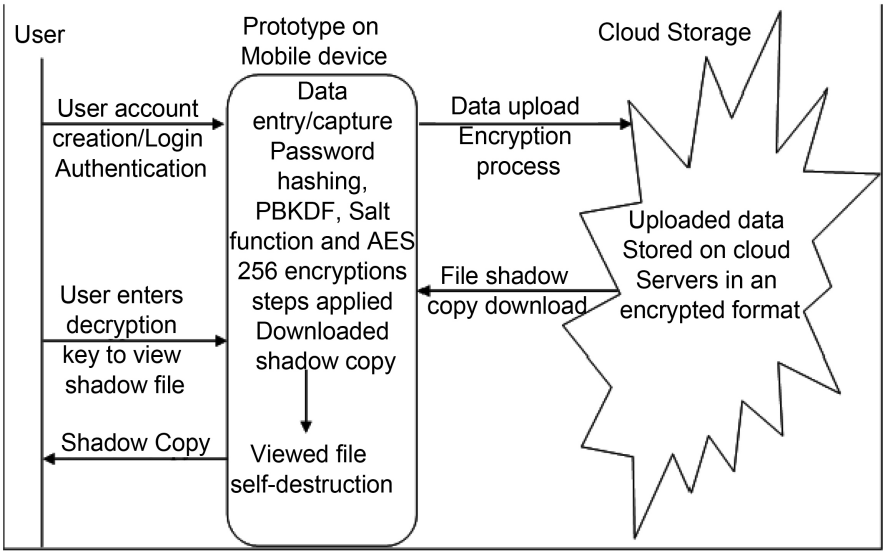

Figure 1. The proposed solution.

solutions will get to be developed with a backend database and servers that gets to contain the encrypted data and encryption keys for self-decryption of the data once it is added to the system. The view of the data will be categorized in groups based on the type of files added, and thus can get to be accessed as a virtual device.

- Automated File Self destruction: The solution will provide the user with self-destruction capabilities. This is whereby the viewed data will be deleted automatically from the device's memory after the user is done with it.
- Additional encryption steps: the solution will apply the AES 256 to encrypt the user data and store it safely. Secondly, the password will be hashed to avoid storing it in plain text. The password based key derivation function will also be used to help reduce vulnerabilities to brute force attacks. Notably, the salt function will be added to help strengthen the encrypted file such that in the event of identical passwords, they would not have the same hash during storage.

4.2. Implementation of the Proposed Prototype and Results

The design tools that were utilized for the design phase included:

- Flow charts: They were used to diagrammatically represent a step by step approach of an algorithm, workflow or process.
- Entity relationship diagrams: This was used to model the database schema to show the relationships between the entities during database design and development.
- Use Case diagrams: The diagrams were used to show the users interaction with the mobile encryption solution.
- Sequence diagrams: This diagram showed how object interact with one another and in the order of interaction.

- Class diagrams: This diagram was used to model the static view of the proposed solution, by showing the system's classes, operations, attributes as well as the relationship that exists among the defined objects.
- Wireframe diagram: This was used for sketching of the Android mobile Application to give a visual representation of the structure and system functionality.

The design had the following security parameters as shown in Figure 2; Automated file destruction capabilities, data storage location, strength of encryption, user login authentication, data recovery in the event of a mobile device loss and access control based on data confidentiality level required.

4.3. Prototype Testing

This section describes the tests that were performed on the developed secure mobile user data encryption tool to ascertain if the set objectives were achieved. It describes the integration tests that were carried out to validate system requirements. During mobile tool development developers sometimes come across validated units that fail to work when combined. Integration tests are carried out

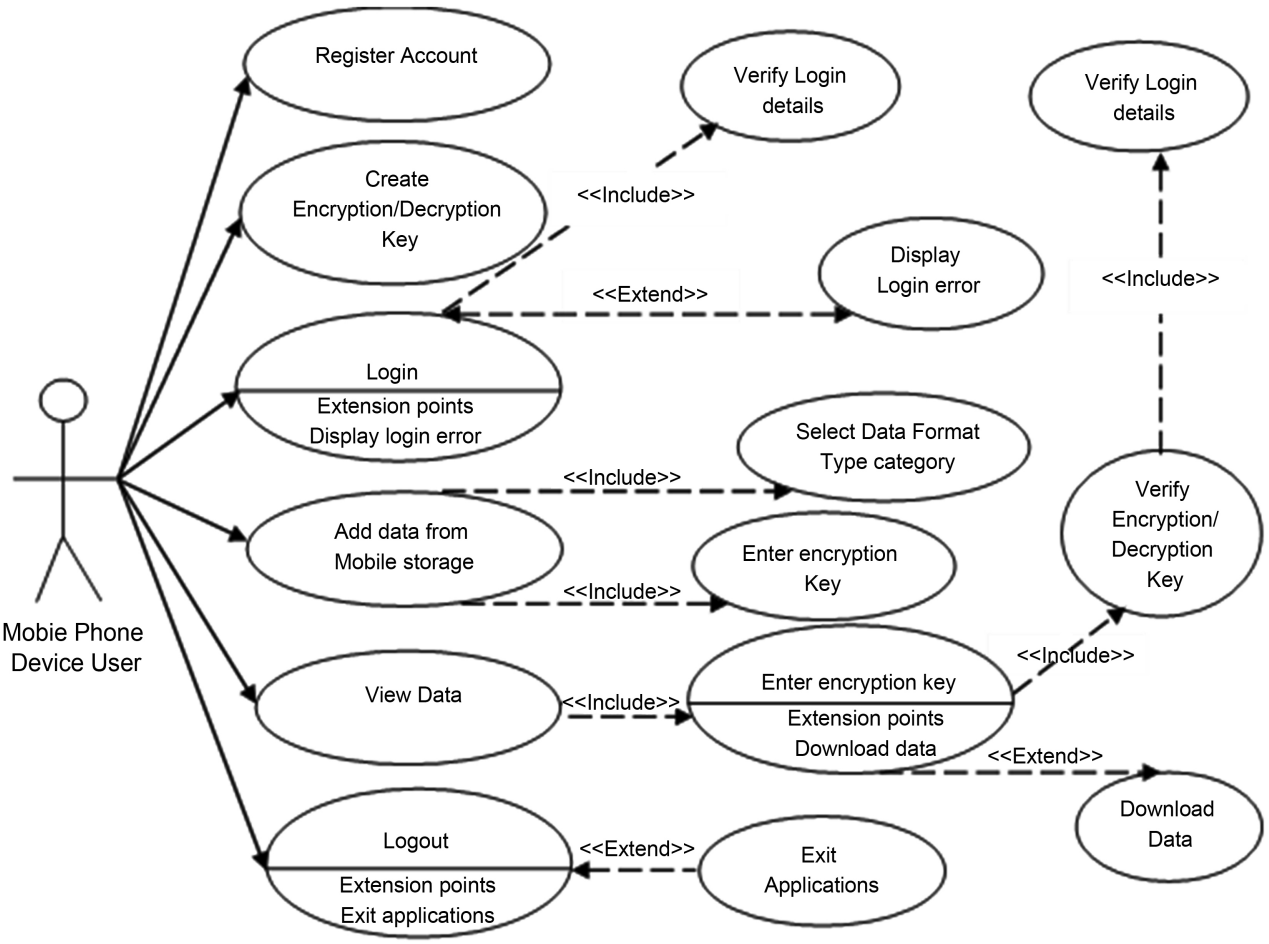

Figure 2. Use-case diagram representing the mobile device user, interactions with the mobile data encryption tool, and the relationships that exist.

to look at the system wide performance, ensuring the functionality between the different units.

Several integration tests were carried out to validate the mobile tool development. These tests involved different functionalities of the system which involved:

Account creation: This integration testing process was performed to ensure that new user account details are captured and verified in the database.

Login connect and authentication: This integration phase was carried out to ensure that the process to connect to the database to verify authentication details was working properly.

Encryption/decryption key creation: This integration testing part was performed to ensure the system allowed for successful creation of encryption/decryption key.

Data upload: This section of the integration testing was conducted to confirm whether the tool was able to successfully upload data.

Download shadow copy: This integration phase was conducted to ensure the system successfully downloaded a shadow copy by validating the entered decryption key.

System logout and automatic file self-destruct: This integration phase was performed to confirm that the tool was able to successfully logout the user and self-destruct the downloaded shadow copy.

After successful completion of integration testing in the described phases, these modules were combined and run hence producing the final product which ran successfully in all of those phases.

4.4. Validation

After successful completion of the prototype testing, the next phase involved its validation. The prototype was validated against the following security parameters: Automated file destruction capabilities, data storage location, strength of encryption, user login authentication, data recovery in the event of a mobile device loss and access control based on data confidentiality level required. Table 2 shows validation findings of the developed prototype.

The results of the validated prototype compared to other existing solutions are as shown in Table 3.

Key:

1) Huawei Silicon chips; Proprietary encryption method.

2) AES 256; A symmetric encryption system that scrambles sensitive data using mathematical calculations to turn data into code.

3) Signal protocol; It amalgamates the Double Ratchet algorithm, Extended Triple Diffie-Hellman (X3DH), pre-keys, and uses AES-256, Curve25519 and HMAC-SHA256 as cryptographic primitives.

4) PBKDF (Password Based Key Derivation Function); used to reduced vulnerabilities to brute force attacks.

5) Salt Function; strengthens the encrypted file such that in the event of identical passwords, they would not have the same hash during storage.

Table 2. Validation findings for the developed prototype.

Security Parameters	Secure cloud storage approach prototype for mobile devices user data Validation	
	Security parameter Factfile	Output Description
Automatic File Self-Destruction capabilities	Yes	The developed prototype was successfully able to self-destruct files after viewing and logout
Data Storage Location	Cloud	The user data was stored in the cloud and no shadow copy was left on the mobile device internal memory
Encryption Method Employed	AES 256, PBKDF, Salt function	The Prototype Successfully applied AES 256 as the main encryption method and the additional security/encryption steps listed
Strength of Encryption	Very High	Based on the encryption steps employed, it made the prototype strength of encryption very high. This is because the used encryption methods are proven to be very difficult to crack by attackers.
Login Authentication required	Yes	The tools requires user login details and successfully authenticates them before login can take place
Data recovery in the event of mobile device loss	Yes	Data is successfully stored in the cloud and can be accessed using a different device with the correct credentials
Access control based on data confidentiality level required	Yes	User is required to create a decryption/encryption key which they are prompted to enter when viewing the uploaded content

Table 3. Analysis of existing mobile devices user data security solutions vs the developed prototype.

	Solutions	Automatic File Self-Destruction capabilities	Data Storage Location	Encryption Method Employed	Strength of Encryption	Login Authentication required	Data recovery in the event of mobile device loss	Access control based on data confidentiality level required
1	Safe for Huawei	No	Mobile device	Huawei HiSilicon chips	AVERAGE	Yes	No	NO
2	Samsung Android Mobile Device Security	No	Mobile device	AES 256	AVERAGE	Yes	No	NO
3	Google cloud storage	No	Cloud	AES 256	HIGH	Yes	Yes	NO
4	Microsoft OneDrive	No	Cloud	AES 256	HIGH	Yes	Yes	NO
5	Signal	No	Cloud	Signal protocol	HIGH	Yes	No	NO

Continued

6	CoverMe	No	Mobile device	AES 128 and AES 256	AVERAGE	Yes	No	NO
7	Secret Box	No	Mobile device	AES 256	AVERAGE	Yes	No	NO
8	Crypt4All	NO	Mobile device and cloud	AES 256	AVERAGE	YES	YES	NO
9	Developed Prototype	Yes	Cloud	AES 256, PBKDF, Salt function	VERY HIGH	Yes	Yes	YES

4.5. Discussion of Results

This section aims to discuss the findings of the research concerning the specific objectives that were set.

As cited by Gkioulos *et al.* (2017) [35], in the field of security focusing on existing mobile devices, these two scholarly work disclose that mobile apparatus owners normally seem to possess high levels of belief in their capabilities to safeguard their mobile devices. Nonetheless, one can say high levels of attentiveness during probable risks are obvious cognizance, and can be apparent for mobile device users. However, they are still uninformed about certain threats as well as remedies that can be used to considerably increase their security. Related observations as well concern mobile device users' behavior, whom usually seem to prefer to prioritize accessing certain mobile tools above protective measures.

Amin *et al.* (2019) [36] recommended a mechanized process that can be used to detect vulnerabilities in mobile devices tools. The key findings achieved in the above-mentioned paper have a corresponding nature to those which are presented in their study as the writers' focuses developing an automated model for identifying faults in mobile tools. The commonalities in those investigations are the elements focused on, which is detection of potential security dangers. Yet, what distinguishes them is that, in Amin *et al.*'s study, key importance is focused on the technological characteristics of security problems according to the run time behavior of a tool.

Mavoungou *et al.* (2016) [37] in their probe, concentrated on attacks and vulnerabilities on mobile networks that represents a major concern for their performance and security. The paper focused on coming up with a list of attacks and at the same time categorizing and classifying them by strongly focusing on attacks based on Internet Protocol, jamming, and signaling. Moreover, in the proposed arrangement of threats, the authors proposed sufficient countermeasures as well as mitigation solutions. Amongst the several deliberated vulnerabilities, they as well showed a compromised mobile device tool security and imperiled user identity privacy as being of utmost significance. Their paper was con-

sidered a technically focused categorization of viable threats to the device network operative. Though, it had matching values to those presented in this thesis, the outburst in the number of mobile device users as well as their security policies affect the overall level of security in general.

Valcke (2017) [38] suggested a broad analysis of the financial industry as well as its faults in the perspective of mobile security. Cybercriminals tend to target such institutions through their mobile tools, as they are avenues for varying security exploitations. The author activists put further urgency on boosting client-side security by using a range of techniques to login, being proactive when it comes to fraud prevention and reinforcement of the client server communication security. In addition to the indicated security challenges, Valcke (2017) [38] emphasizes the role of mobile tools developers who ought to be keen when it comes to the security aspect of mobile tools, whereas, respecting user experience protocols. The relationships between both of these studies view the behavior of the user as a vital mobile security factor. The authors' statement is the perfect paper recapitulation: "You still have to balance security with ease of use, and you still have to ensure that your core business logic is not subject to any exploits too."

Hatamian *et al.* (2019) [39] in his research, primarily centered on mobile tools inventors as the original deterrent mechanism from attacks, fraudulent activities or threats targeting mobile device users. The writer suggests "a security or privacy design guide catalogue for mobile tools developers to help them understand and adopt only applicable practices that advance security or privacy in smartphone tools." The author also directs to the innovators as the people that can fulfil privacy and security principles, thereby becoming an extremely crucial aspect in mobile security requirements preservation.

Mobile gadgets are usually smaller in size, meaning misplacing or losing them is easy. For that reason, the physical security which is currently a key concern for mobile devices ought to be taken into consideration. The utmost clear-cut issue is not just losing the device itself but as well the data stored in its Read Only Memory and any extra approvals used to retrieve information from both external and internal sources (for instance; e-mail, bank account details, corporate intranet, as well as social media). With respect to that information, the risk involved usually high and relates to ransom demands and blackmail, focused against organizations and individuals. This paper has discussed the countermeasures that can be applied so as to mitigate related risks in the event one loses the device. Nonetheless, with the continuing growth of cyber attacker contrivance, all-rounded ways that can guarantee security of mobile devices from those menaces do not exist.

Combination of modern efforts by the business communities and exploration in the extent of mobile security excelled in development of revolutionary remedies by adapting and adopting AES 256 and blowfish, but in comparison to achieving both speed and efficiency while getting the job done, AES 256 comes

out to be the most appropriate algorithm.

It also can be noted that mobile security threats and existing solutions are relatively the same globally, while the policy management for the security remains local, usually, fixed to divergent business-related occurrences and mobile tool milieu. From the collected, analyzed and validated results of this study, we can determine that explicit knowledge symbolizes comprehension of the generative undertakings that make up the basis of mobile safety and data security.

The practical and conceptual repercussions concern the recognition and analysis of several issues, with regard to the threats and existing solutions in the mobile security field. Additionally, the research findings pave way for other indagating areas worth considering by practitioner and academia factions.

On the other hand, study outcomes indicate factual corroboration that shows both best practices and threats that presently subsist in the extent of mobile data storage security. Substantial validity and reliability of the results necessitate the participation of security specialists in a larger number to confirm the key study findings and explore in-depth the problem with separate groups of users in a quantifiable method.

The developed tool's security parameters were tested in which it was noted that for confidentiality, all the confidential user data were stored in an encrypted format using the key provided by the user to encrypt it, thus the files would not be able to be accessed by any unauthorized person. For the storage of the key and password created by the user, all the information was stored in a server using the Secure Hash Algorithm (SHA3) hash algorithm and combined with salt to make it more secure during storage. Encryption of data was carried out using AES 256 Algorithm which is a very strong algorithm considered extremely difficult to crack by hackers. For the availability of data, the files were encrypted and uploaded to a cloud-based server from which they can be accessed wherever they are needed.

The modern security technology advancements, available from either the level of the application or OS, expose fewer weaknesses or vulnerabilities. Consequently, complete security is dependent on the behavior demonstrated by users and efforts undertaken arbitrarily. Comprehension of the issues influencing users' motivations or intentions to use and accept specific technologies is vital to increasing privacy and security concerns at the individual level.

5. Conclusions

The data gathered at the requirement gathering and analysis stage gave vital statistics that was used in the implementation of the system. The system design aided in the development of the secure mobile user data encryption tool in ensuring that the developed tool met the research objectives of the study. Various tests were carried out on the tool to validate the system requirements. The prototype was validated by comparing its security features against those for similar existing solutions. These security parameters included: Automated file destruc-

tion capabilities, data storage location, strength of encryption, user login authentication, data recovery in the event of a mobile device loss and access control based on data confidentiality level required. From the above indicated security parameters, the following key findings were identified after validation:

The developed prototype was successfully able to self-destruct files after viewing and logout. The user data was stored in the cloud and no shadow copy was left on the mobile device internal memory. The Prototype successfully applied AES 256 as the main encryption method and the additional security/encryption steps listed. Based on the encryption steps employed, it made the prototype strength of encryption very high. This is because the used encryption methods are proven to be very difficult to crack by attackers. The tools require user login details and successfully authenticate them before login can take place. Data is successfully stored in the cloud and can be accessed using a different device with the correct credentials. User is required to create a decryption/encryption key which they are prompted to enter when viewing the uploaded content.

The research enables mobile device manufacturers to leverage on the solution and patch their systems in order to improve on mobile device storage security. It also creates awareness to mobile device users to be more vigilant by highlighting the existing threats and vulnerabilities.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Pearlson, K.E., Saunders, C.S. and Galletta, D.F. (2019) Managing and Using Information Systems: A Strategic Approach. John Wiley & Sons, Hoboken.

[2] Haynes, M.N. (2015) Systems, Devices, and/or Methods for Managing Information.

[3] Ramalingam, M., Mathews, S.S. and Inbaraj, J. (2017) Mobile Device Security System.

[4] Aldossary, S. and Allen, W. (2016) Data Security, Privacy, Availability and Integrity in Cloud Computing: Issues and Current Solutions. *International Journal of Advanced Computer Science and Applications*, **7**, 485-498. https://doi.org/10.14569/IJACSA.2016.070464

[5] Yalew, S.D. (2018) Mobile Device Security with ARM TrustZone.

[6] Mollah, M.B., Azad, M.A.K. and Vasilakos, A. (2017) Security and Privacy Challenges in Mobile Cloud Computing: Survey and Way Ahead. *Journal of Network and Computer Applications*, **84**, 38-54. https://doi.org/10.1016/j.jnca.2017.02.001

[7] Gupta, S., Buriro, A. and Crispo, B. (2018) Demystifying Authentication Concepts in Smartphones: Ways and Types to Secure Access. *Mobile Information Systems*, **2018**, Article ID: 2649598. https://doi.org/10.1155/2018/2649598

[8] Padma, B. and Kumar, G.R. (2016) A Review on Android Authentication System Vulnerabilities. *International Journal of Modern Trends in Engineering and Research* (*IJMTER*), **3**, 118-123. https://doi.org/10.21884/IJMTER.2016.3015.8PEUS

[9] Murugesan, S. (2019) The Cybersecurity Renaissance: Security Threats, Risks, and Safeguards.

[10] Franchi, E., Poggi, A. and Tomaiuolo, M. (2015) Information and Password Attacks on Social Networks: An Argument for Cryptography. *Journal of Information Technology Research*, **8**, 25-42. https://doi.org/10.4018/JITR.2015010103

[11] Kumari, S. (2017) A Research Paper on Cryptography Encryption and Compression Techniques. *International Journal of Engineering and Computer Science*, **6**, 20915-20919. https://doi.org/10.18535/ijecs/v6i4.20

[12] Heath, S. (2018) Methods and/or Systems for an Online and/or Mobile Privacy and/or Security Encryption Technologies Used in Cloud Computing with the Combination of Data Mining and/or Encryption of User's Personal Data and/or Location Data for Marketing of Internet Posted Promotions, Social Messaging or Offers Using Multiple Devices, Browsers, Operating Systems, Networks, Fiber Optic Communications, Multichannel Platforms.

[13] Busch, M., Westphal, J. and Mueller, T. (2020) Unearthing the Trusted Core: A Critical Review on Huawei's Trusted Execution Environment. 14*th Workshop on Offensive Technologies*, August 11, 2020. https://dl.acm.org/doi/abs/10.5555/3488877.3488881

[14] Ning, Z. and Zhang, F. (2019) Understanding the Security of Arm Debugging Features. 2019 *IEEE Symposium on Security and Privacy* (*SP*), San Francisco, 19-23 May 2019, 602-619. https://doi.org/10.1109/SP.2019.00061

[15] Zhang, L. and Chen, J. (2018) Encryption Method, Decryption Method, and Related Apparatus.

[16] Park, M., Kim, G., Park, Y., Lee, I. and Kim, J. (2019) Decrypting Password-Based Encrypted Backup Data for Huawei Smartphones. *Digital Investigation*, **28**, 119-125. https://doi.org/10.1016/j.diin.2019.01.008

[17] Kanonov, U. and Wool, A. (2016) Secure Containers in Android: The Samsung KNOX Case Study. *Proceedings of the* 6*th Workshop on Security and Privacy in Smartphones and Mobile Devices*, Vienna, 24 October 2016, 3-12. https://doi.org/10.1145/2994459.2994470

[18] Lu, Y.-F., Kuo, C.-F. and Fang, Y.-Y. (2016) Efficient Storage Encryption for Android Mobile Devices. *Proceedings of the International Conference on Research in Adaptive and Convergent Systems*, *RACS* 2016, Odense, 11-14 October 2016, 213-218. https://doi.org/10.1145/2987386.2987418

[19] Henziger, E. and Carlsson, N. (2019) The Overhead of Confidentiality and Client-Side Encryption in Cloud Storage Systems. *Proceedings of the IEEE/ACM International Conference on Utility and Cloud Computing*, Auckland, December 2019, 209-217. https://doi.org/10.1145/3344341.3368808

[20] Jayapandian, N. and Md Zubair Rahman, A. (2018) Secure Deduplication for Cloud Storage Using Interactive Message-Locked Encryption with Convergent Encryption, to Reduce Storage Space. *Brazilian Archives of Biology and Technology*, **61**, e18160609. https://doi.org/10.1590/1678-4324-2017160609

[21] Jayapandian, N., Rahman, A.M.Z., Radhikadevi, S. and Koushikaa, M. (2016) Enhanced Cloud Security Framework to Confirm Data Security on Asymmetric and Symmetric Key Encryption. 2016 *World Conference on Futuristic Trends in Research and Innovation for Social Welfare* (*Startup Conclave*), Coimbatore, 29 February 2016-1 March 2016, 1-4. https://doi.org/10.1109/STARTUP.2016.7583904

[22] Chu, C.-K., Zhu, W.-T., Han, J., Liu, J.K., Xu, J. and Zhou, J. (2013) Security Concerns in Popular Cloud Storage Services. *IEEE Pervasive Computing*, **12**, 50-57. https://doi.org/10.1109/MPRV.2013.72

[23] Gelb, D.K. (2014) Using Technology to Prepare for Trial. *GPSolo*, **31**, 20.

[24] Lonsky, R. (2018) Security of Microsoft OneDrive.

[25] Ferdaus, A.A., Yousuf, M.A.I. and Haque, M.A. (n.d.) Cloud Storage.

[26] Ermoshina, K., Musiani, F. and Halpin, H. (2016) End-to-End Encrypted Messaging Protocols: An Overview. *Third International Conference*, *INSCI* 2016—*Internet Science*, Florence, September 2016, 244-254. https://doi.org/10.1007/978-3-319-45982-0_22

[27] Cohn-Gordon, K., Cremers, C., Dowling, B., Garratt, L. and Stebila, D. (2020) A Formal Security Analysis of the Signal Messaging Protocol. *Journal of Cryptology*, **33**, 1914-1983. https://doi.org/10.1007/s00145-020-09360-1

[28] Cohn-Gordon, K., Cremers, C., Dowling, B., Garratt, L. and Stebila, D. (2017) A Formal Analysis of the Signal Messaging Protocol. *Journal of Cryptology*, **33**, 1914-1983. https://doi.org/10.1109/EuroSP.2017.27

[29] Abu-Salma, R., Sasse, M.A., Bonneau, J., Danilova, A., Naiakshina, A. and Smith, M. (2017) Obstacles to the Adoption of Secure Communication Tools. 2017 *IEEE Symposium on Security and Privacy*, San Jose, 22-24 May 2017, 137-153. https://doi.org/10.1109/SP.2017.65

[30] Zhang, X., Baggili, I. and Breitinger, F. (2017) Breaking into the Vault: Privacy, Security and Forensic Analysis of Android Vault Applications. *Computers & Security*, **70**, 516-531. https://doi.org/10.1016/j.cose.2017.07.011

[31] Schneier, B., Seidel, K. and Vijayakumar, S. (2016) A Worldwide Survey of Encryption Products. Berkman Center Research Publication No. 2016-2. https://doi.org/10.2139/ssrn.2731160

[32] Bursać, M., Vulović, R. and Milosavljević, M. (n.d.) Comparative Analysis of the Open Source Tools Intended for Data Encryption.

[33] Ritacco, A. and Wills, C. (2018) Peering into the Home Network.

[34] Luangoudom, S., Nguyen, T., Tran, D. and Nguyen, L.G. (2019) End to End Message Encryption Using Poly1305 and XSalsa20 in Low Power and Lossy Networks. 11*th International Conference on Knowledge and Systems Engineering* (*KSE*), Da Nang, 24-26 October 2019, 1-5. https://doi.org/10.1109/KSE.2019.8919479

[35] Gkioulos, V., Wangen, G., Katsikas, S.K., Kavallieratos, G. and Kotzanikolaou, P. (2017) Security Awareness of the Digital Natives. *Information*, **8**, Article No. 42. https://doi.org/10.3390/info8020042

[36] Amin, A., Eldessouki, A., Magdy, M.T., Abdeen, N., Hindy, H. and Hegazy, I. (2019) Androshield: Automated Android Applications Vulnerability Detection, a Hybrid Static and Dynamic Analysis Approach. *Information*, **10**, Article No. 326. https://doi.org/10.3390/info10100326

[37] Mavoungou, S., Kaddoum, G., Taha, M. and Matar, G. (2016) Survey on Threats and Attacks on Mobile Networks. *IEEE Access*, **4**, 4543-4572. https://doi.org/10.1109/ACCESS.2016.2601009

[38] Valcke, P. (2017) EU Policy on Telecommunications and Electronic Communications-7. Open Internet Access ("Net Neutrality") & Roaming Regulation. Training for the European Parliament Organized by the College of Europe, Brussels.

[39] Hatamian, M., Serna, J. and Rannenberg, K. (2019) Revealing the Unrevealed: Mining Smartphone Users Privacy Perception on App Markets. *Computers & Security*, **83**, 332-353. https://doi.org/10.1016/j.cose.2019.02.010