Enhancement of Decryption Operation for Resource Limited Devices

Kosmas Kapis
Department of Computer Science and Engineering
CoICT
University of Dar es Salaam, UDSM
Dar es Salaam, Tanzania
E-mail: kkapis [AT] gmail.com

Marseline Michael Mtey
Department of Information Technology
Faculty of Science and Education
University of Iringa, UoI
Iringa, Tanzania

Abstract - **The computing capability of resource-limited devices cannot be compared to that of computers when a higher resource demanding applications is to be executed. With exceptional to laptop computers, mobile devices for example, are less powerful in computations like video streaming, teleconferencing and even decrypting information. This forces users to outsource those operations including even key generation and storage to third party service providers like cloud servers.**

However the common problem of cloud servers and other third party service providers is the breach of confidentiality. This may disclose sensitive information to unauthorized users which affects privacy. Other problems include the communication latency with clients, the security protocols overhead, and the intrusion through data access patterns. To address those challenges, key generation by data owner and a partial decryption outsourcing to a cloudlet as a local cloud server was proposed. These help to solve the breach of confidentiality and latency problems. In addition, the minimum computation overhead is assured by use of light encryption protocols known as Attribute Based Encryption (ABE). This protocol uses normal public key cryptography functions with additions of users' attributes.

Through synthesized experiments the devised solution was compared with the existing solutions and the results were analyzed. It was revealed that with enhanced decryption the latency was dropped by 0.55s which is 23% reduction. Also the use of Private Information Retrieval (PIR) technique ensured that any unauthorized intruders cannot have access to the data.

Keywords - Security, Decryption, Intrusion, Confidentiality, Communication latency, Attribute Based Encryption, Mobile-Cloud Computing.

I. INTRODUCTION

Recently, hand held mobile devices such as smart phones and phone tablets have increasingly became powerful due to their web browsing capabilities on top of the voice oriented function (1). This domination has been boosted by the introduction of web 2.0 and 3.0 technologies which have enabled the social and data sharing capabilities to these devices (2). However, despite of these progress, they still cannot compete with desktop computers in executing high resource demanding applications due to the possession of limited resources (3).

To adhere to this problem, the technique of offloading resource demanding computations to third party service providers was proposed (4). This process identifies and transfers the resource-demanding computations of mobile devices to cloud-based resources (5). In cloud computing servers, the computing resources are delivered as a service which is the current iteration of utility computing though which users can rent resources (Li, Li, Chen, Jia, & Lou, 2015).

On the other hand, a light cryptographic technique was used in order to reduce the security protocol overhead. An Attribute Based Encryption protocol (ABE) which uses the normal public key cryptography functions with addition of users' attributes was suggested.

Among the related studies that use ABE is an Attribute-based Access to Scalable Media where the fully outsourcing process returns plaintext to smartphone users (7). This study proposed the access and offloading of content from smartphones to cloud servers which run on virtual machine. In addition to that (8) came up with a technique to access health care records using mobile phone with decryption process outsourced to cloud servers. Apart from (4),(7) and (8) which proposed outsourcing operations to cloud server, other related works include (13) and (14). In all these works the cloud servers were proposed without considering the latency problem. (15) and (16) noted this and proposed outsourcing operation to cloudlet and edge computers respectively.

With outsourcing operations, the resource limitation burden of resource limited devices is solved. Despite of this advantage, the outsourcing operation may become severe due to the fact that the included parties are run by third party service providers. Due to this, it was noted that there might be a communication latency between the resource limited device and a cloud server.

Another noted challenge is a breach of data confidentiality is which is described to be exercised by those parties (9). This breach is caused by the curiosity behavior of the third party service providers like cloud servers during their operation. They often try to find out much information about their operation whether it is fully or partially outsourced to them. An evidence of breach were noted from 2007 to 2009 in social networks like Salesforce.com, Google and Twitter (10). In those breaches the attackers used the gained privilege to access confidential information without the knowledge of the owners.

Considering those challenges, strong measures must be taken during computation outsourcing in order to avoid the breach of confidentiality and communication latency. Therefore this study considered all those challenges and other obtained from literatures to devise an alternative solution to the problem. Later the solution was simulated and tested in order to understand its usefulness and contribution.

A. Abbreviations and Acronyms

MCC	*Mobile Cloud Computing*
MCP-ABE	*Multipletext Ciphertext Policy Attribute Based Encryption*
PaaS	*Platform as a Service*
PDA	*Personal Digital Assistant*
PHR	*Personal Health Records*
TA	*Trusted Authority*

II. METHODOLOGY

This paper was conducted based on experimental research design in order to have control of the research variables. It followed the laboratory setting by only simulating the solution while leaving the actual implementation for future work.

III. EXPERIMENTS

Through simulations, both active and passive measurements values were recorded. Synthesized experiments were conducted for different variables to investigate the ways they were affected by various extreme conditions (11).

A. Request processing paths

In order to get the output of the experiments the associated data phases were isolated and readings at each phase were noted. These included clients, data owner, and cloudlet.

Initially the responses from the data owner resulted from clients requests were measured at different stages in order to compare the time delay. With data owner as a trusted authority and later with cloudlet, the result were noted. The time taken for the clients request to be responded trough the two options were noted. With full decryption and key generation by the cloudlet, the results were noted. Also the same results were taken when the operation was done partially by the cloudlet while the associated keys being generated by the data owner. The total time for the two options were taken and compared with different data values associated with each request. The aim was to see the effect of the computations time by the cloudlet with its request reply phenomena.

The computations, propergations and queing delays of the request and responce through the network were measured. With that being noted, at various points, the operations associated by clients, data owners and third party servers can be categorised as follows.

B. The Encrypt Phase

In this phase the data owners encrypt the information before delegating it to the third party service providers. Encrypting the information safeguards it against the malicious users when stored in the third party servers or when is transmitted.

C. The Setup Phase

This phase is where the Public Parameters (PK) and a Master Key (MK) are generated. In this phase the algorithm takes the implicit security parameter to output those public parameters and a master key. These are the input for the keyGen phase which is another phase in the data sharing process.

D. The KeyGen Phase

In the key generation phase, a trusted party is chosen and given the responsibility. This trusted authority is responsible for generation of keys used for the communicating parties. It generate private key and a transformation key for the users which are used to encrypt and decrypt the shared information.

E. The Transform Phase

This phase is the one at which the cloud server or a cloudlet performs partial decryption operation on the encrypted data. It transforms the ciphertext into a simple ciphertext (partially decrypted) before sending it to users. If the user's attribute satisfies the access structure associated with the ciphertext he/she uses the decrypt phase to recover the plaintext from the transformed ciphertext.

F. The Decrypt Phase

As the Transform algorithm transforms the ciphertext into a simple cipher, lastly the client uses the decrypt phase to recover the plaintext from the transformed ciphertext.

III. THE MEASUREMENTS

The measuments at each points of path through which the request reaches, constitutes to the passive measurements value. Also the actual data collected at those points gives the actual passive measurements values. On the other hand the values recorded from the synthesized experiments constituted to the active measurements.

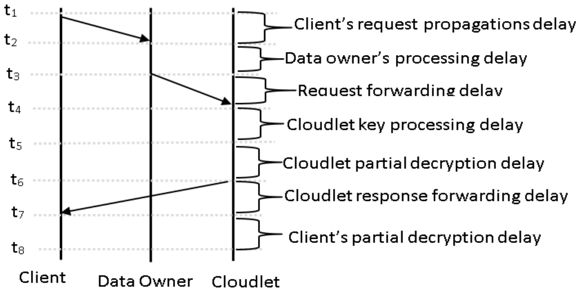
t1
t2
t3
t4
t5
t6
t7
t8
Client's request propagations delay
Data owner's processing delay
Request forwarding delay
Cloudlet key processing delay
Cloudlet partial decryption delay
Cloudlet response forwarding delay
Client's partial decryption delay
Client
Data Owner
Cloudlet

Figure 1. Client, Data Owner and Cloudlet Interactions

A. Passive Measurements

Figure 1 shows various delay components that constitute to the total time taken between the users request and reponce. At each phase the processing that takes place contribute to the total time taken for the user's request to be responded. The components t_2–t_1, t_3–t_2, t_4–t_3, t_5–t_4, t_6–t_5 and t_7–t_6 represent the setup, the key generation, the propagations and the processing delays. These are contributed by interactions between the client, cloudlet and the data owner. These delay components of the devised solution are illustrates in the following sub-sections.

1) *Clients request propagations delay (t_{cp})*

The delay associated with the clients request starts with the time taken to reach the data owner. The delay component $t_{cp}= t_2 – t_1$ is the time taken from the request start time to the time when the data owner receives it.

2) Data owners' request processing delays tdp and tdpr)

The delay components associated with the setup phase, and the keyGen phase also contribute to the time taken for the request to be responded. The time gap $t_{dp} = t_3 – t_2$ is the delay component for processing the request by the data owner before forwarding it to the cloudlet. Also the delay component $t_{dpr}= t_4$–t_3 is the propagation delay for the request forwarded to the cloudlet by the data owner.

3) Cloudlet's computation delay (tcpd)

The delay component $t_{cpd}= t_5$–t_4 is the delay component associated with the partial decryption operation done by the cloudlet. This is when the ciphertext is transformed into a simple cipher (partially decrypted) before sending it to clients. This constitutes to the transform phase delay which is among the associated delay components. Furthermore the delay component associated with forwarding the response to the client which is $t_{cpr} = t_7$-t_6 also contributes to the total time.

4) Client's transformed text decryption delay (tcdp)

On receiving the transformed text from the transform phase, the client completes the remaining decryption process in order to disclose the plaintext. The delay component associated with this process is $t_{cdp}=t_6$-t_5 which is the time taken to obtain the plaintext from the partially decrypted information by the cloudlet.

B. Active Measurements

In order to collect data from multiple points, various synthesized experiments were conducted in order to get the actual values of those points. Results from both a cloud server and a cloudlet were taken for comparison reasons in order to validate the importance of using cloudlet. Active requests were sent across both functional points in the form of synthetic requests. The intension of those requests was to understand how they affect the clients' request-forwarding and response processing.

From the measurements, the clients request forwading constitute to the request forwading delay. The associated time components for the forwarding delay are $t_{cp} = t_2 – t_1$, $t_{dp} = t_3 – t_2$, and $t_{dpr} = t_4$–t_3. On the other hand, the time components $t_{cpd} = t_5 – t_4$, $t_{cdp} = t_6 - t_5$ and $t_{cpr} = t_7 - t_6$ are the delay components associated with the response forwarding from the cloudlet or cloud server to the client.

IV. RESULTS

From three different experiments with files ranging from 1kilobyte (kb) to 1megabyte (mb) the results were noted and compared as shown.

A. *Using cloud server as fully outsourced third party*

To start with the delay components associated with cloud server full decryption and full key management with the synthesized request for a videos of 1kb to 1Mb size, was as follows:

$t_{cp}= t_2 – t_1 \rightarrow 0.04$ milliseconds which is the time for the request to be received by the data owner. Another delay component for the setup and initialization process by the data owner which is $t_{dp} = t_3 – t_2 \rightarrow 0.03$ milliseconds. From there the request Id and the setup parameters for keyGen phase are forwarded to the cloud server with the delay component $t_{dpr} = t_4$–$t_3 \rightarrow 0.05$ms. At the cloud server the keyGen and the full decryption operation constitute to the delay component of $t_{cpd}= t_5$–$t_4 \rightarrow 0.09$ms which is the delay combination of the two processes. The decrypted information is then forwarded to the client with a delay component $t_{cpr}=t_6$-$t_5 \rightarrow 0.02$ms through the network.

On receiving the plaintext the client will have no heavy task rather than normal Id verification before accepting the response. A small delay associated with the process which is $t_{cdp}=t_8$-$t_7 \rightarrow 0.016$ms is almost negligible. The total time taken for the files to be downloaded to the client with the synthesized request was 0.25 ms for clouds' full decryption and full key management process.

B. *Using cloudlet as a fully outsourced third party*

Here the same experiment was carried out on the same data value and operations with only the cloud server replaced by a cloudlet. The delay components associated with this experiment were as follows:

i. $t_{cp}= t_2 - t_1 \rightarrow 0.04$ms.

ii. $t_{dp} = t_3 - t_2 \rightarrow 0.03$ms.

iii. $t_{dpr}= t_4 - t_3 \rightarrow 0.05$ms.

iv. $t_{cpd}= t_6 - t_4 \rightarrow 0.09$ms.

v. $t_{cpr}=t_7-t_6 \rightarrow 0.026$ms.

vi. $t_{cld} = t_8-t_7 \rightarrow 0.02$ms

The total time taken for the files to be downloaded to the client with the synthesized request was 0.25 milliseconds for cloudlets' full decryption and full key management process.

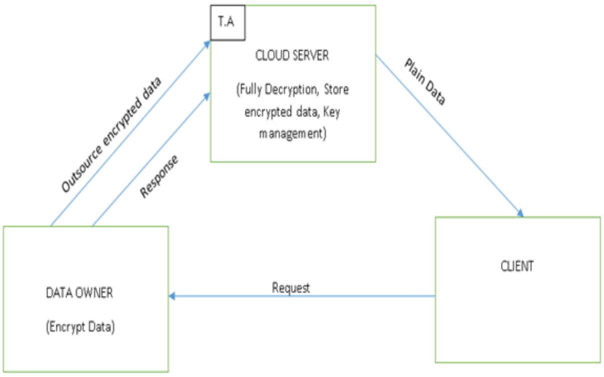

Figure 2. Components of existing solution

C. *Using cloudlet as partial outsourced third party*

Lastly with a cloudlet as a third party to outsource partial decryption the delay components associated with this experiment were as follows:

i. $t_{cp}= t_2 - t_1 \rightarrow 0.04$ms.

ii. $t_{dp} = t_3 - t_2 \rightarrow 0.038$ms.

iii. $t_{dpr}= t_4 - t_3 \rightarrow 0.029$ms.

iv. $t_{cpd}= t_6 - t_4 \rightarrow 0.04$ms.

v. $t_{cpr}=t_7-t_6 \rightarrow 0.03$ms.

vi. $t_{cld}= t_8-t_7 \rightarrow 0.03$ms

The total time taken for the files to be downloaded to the client with the synthesized request was 0.21 milliseconds for cloudlets' partial decryption with full key management process carried by the data owner.

V. THE DEVISED SOLUTION

From the existing solution this part derived to a solution that eliminates some of the challenges reviewed from the literatures.

Figure 3 shows the components of the existing solution that contributes to the complete path from the client to the cloud server through the data owner. The tasks of a cloud server is to store encrypted data, to perform full decryption and key generation.

Due to the sensitivity and confidentiality of the outsourced data it was also noted that the plaintext from the cloud server to the client may invite intruders. An example is the man in the middle attack which is another challenge considering the previously noted latency and breach of confidentiality problems.

To avoid those challenges, the devised solution assigned the key management process to the data owner. Also the techniques of anonymizing the identity of users was the adopted in order to protect the data access pattern. A Priviate Information Retrieval (PIR) technique that protects users' read pattern was used (Nair, P, & Kumar, 2015). This ensures that intruders cannot access the unauthorized data by learning the access pattern.

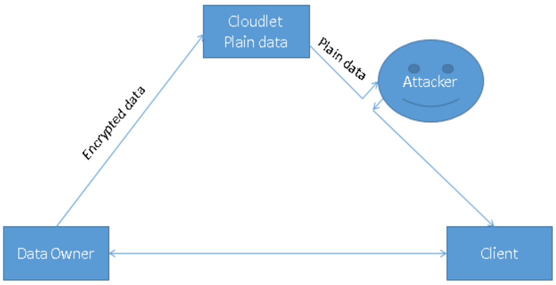

Figure 3. A system without data access control

A systems without data access control is shown in Figure 4. In this system it is obvious that a man in the middle attack can happen. This problem was solved by adopting the PIR technique as shown in figure 5.

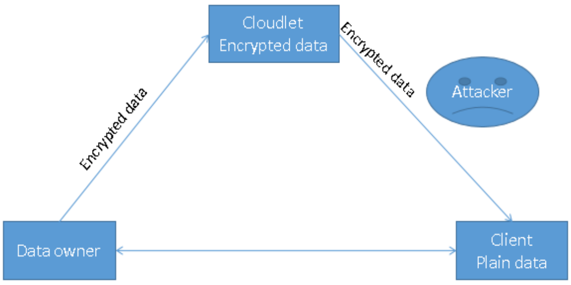

Figure 4. A system with data access control

The final comparisons between the existing solution and the devised solution is shown in figure 6. Here different files were decrypted and the time taken from the request to the response were noted and plotted in the graph.

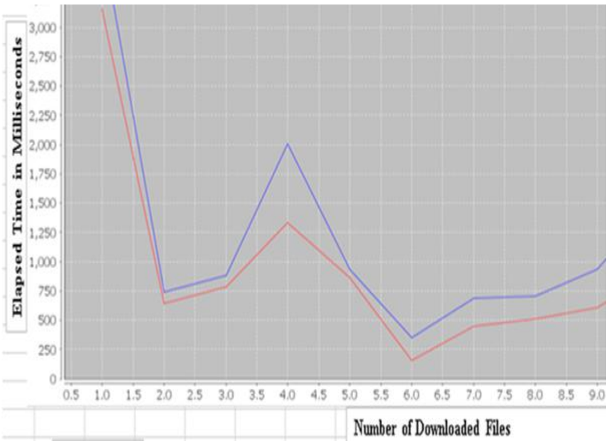

Figure 5. Files decrypted vs. elapsed time

Key:

Red Line = Output in the devised solution

Blue Line = Output in the existing solution

VI.DISCUSSION

In devising the solution, from various literatures, the operational strengths and weaknesses of available solutions were analyzed. These helped to discover the deep insight of the study and identify the challenges and a gap. From the literature it was found that most of them contributed to some extent in enhancing the decryption operation for resource limited devices. However there are various areas that were not covered by most of them including the failure to choose an optimal mechanism to decide which application is to be deployed in cloud, and which one in the local device. Also the latency between mobile and cloud communications, the breach of confidentiality by cloud servers and the system intrusion through users' information access pattern were other challenges.

From the observed challenges an alternative solution was devised which made sure that the challenges encountered during the literature survey were dealt with. A solution with partial decryption outsourcing to a cloudlet, which is a local cloud server at a close proximity to clients was devised. This reduced the latency of communication between the cloud server and clients. In addition to that the partial decryption outsourcing ensures that the cloudlet gains no important detail of the outsourced data and information. On top of that, users' data access pattern was controlled by the use of PIR techniques to protect the read operations against intruders.

Lastly the solution was evaluated in order to test its usefulness and contributions. By taking the comparisons between data decrypted by clients from the cloud server and cloudlet the comparisons between their latency were done. Also the totally outsourced decryption and partially outsourced decryption were compared, and the time gaps were noted down at different loads of the systems.

On top of that, the comparison of simulation between the computation time of an AMD A8-4500M APU – Radeon (tm) HD, Samsung laptop with 1.9 GHz, 8.00 GB and 64-bit Operating System as a cloud server and that of JAVA FX touch phone as a client were noted. The result revealed beyond doubt that secure partial offloading of decryption operation helps in reducing the computation burden for resource limited devices especially in mobile cloud computing.

After all the comparisons and measurements a solution based on challenges is shown in figure 6. It shows the components of the devised solution with all the security risks precautions implemented.

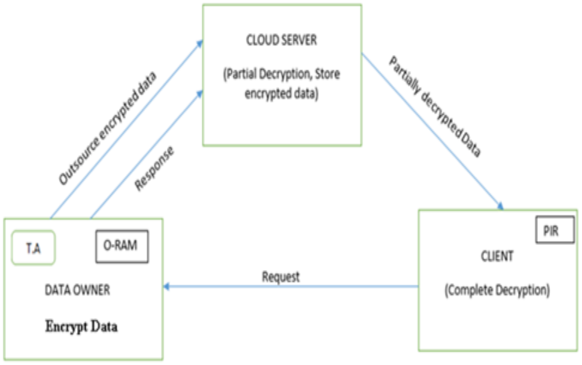

Figure 6. Components of the Devised Solution

VII. CONCLUSIONS

This paper focused on reducing the decryption burden of resource limited devices that use cloud computing as a third party service provider. It mainly aimed at decreasing the decryption time, memory usage hence reducing the power consumption which is a problem for smartphones. The research successfully reduced the decryption time by 23% by comparing with the existing solution. Also by using the PIR technique, the user's data access pattern mainly the read operation was protected against man in the middle attack. This solution was proved helpful when evaluated which can be implemented in actual working environment to enjoy its usefulness in resource limited devices.

VIII. FUTURE WORKS

Despite the challenges encountered, there are also opportunities found in this study which need to be cultivated. The first opportunity is the implementation of the devised solution in order to test it in actual working environment. Another future work to be done is for the smartphone manufacturers to include built-in applications that ensure the data read operation is secure rather than depending on the external applications like PIR.

ACKNOWLEDGEMENTS

The authors would like to thank the teaching staffs of the University of Dar es Salaam especially Dr. Kosmas Kapis for his support in completing this paper. Also the authors would like to thank University of Iringa management and staffs for their technical support and duties exemption during the time of conducting this research. Without their support, I wouldn't have time to complete this paper.

REFFERENCES

[1] Malligai, V., & Kumar, V. V. (2014). Cloud Based Mobile Data Storage Application System. International Journal of Advanced Research in Computer Science and Technology, 2(March), pp.126–128.

[2] Fuchs, C. (2010). Social Software and Web 2.0: Their Sociological Foundations and Implications. In Handbook of Research on Web 2.0, 3.0 and X.0: Technologies, Business, and Social Applications Vol. I, pp. 764–789.

[3] Antero, J., Matti, K., & Sakari, L. (2012). Mobile computation offloading-Factors affecting technology evolution. In 11th ICMB pp. 137–148.

[4] Lin, H., Shao, J., Zhang, C., & Fang, Y. (2013). CAM : Cloud-Assisted Privacy Preserving Mobile Health Monitoring. IEEE Transactions on Information Forensics and Security, 8(6), pp. 1–17.

[5] Aminzadeh, N., Sanaei, Z., & Ab Hamid, S. H. (2015). Mobile storage augmentation in mobile cloud computing: Taxonomy, approaches, and open issues. Simulation Modelling Practice and Theory, 50, pp. 96–108.

[6] Li, J., Li, J., Chen, X., Jia, C., & Lou, W. (2015). Identity-based Encryption with Outsourced Revocation in Cloud Computing. IEEE Transactions on Computers, 64(2), pp.425–437

[7] Wu, Y., Wei, Z., & Deng, R. H. (2013). Attribute-based Access to Scalable Media in Cloud-assisted Content Sharing Networks. IEE Transaction on Multimedia: Cloud Based Mobile Media: Infrastructure, Service and Application, pp. 1–29.

[8] Ganesan, K., & Vijayakumaran, C. (2014). HealthCare Monitoring Solution with Decryption Outsourcing by Parallel Computing in Cloud. International Journal of Innovative Research in Computer and Communication Engineering, 2(1), pp. 56–64.

[9] Yu, S. (2010). Data Sharing on Untrusted Storage with Attribute-Based Encryption. Worcester Polytechnic Institute.

[10] Kumar, K., & Lu, Y. (2010). Cover Feature Cloud Computing For Mobile Users: Computation Save Energy? IEEE Computer Society, pp. 51–56.

[11] Berndtsson, M., Hansson, J., Olsson, B., Lundell, B., Jörgen Hansson, M. B., & Björn Olsson, B. L. (2008). Thesis Projects. A Guide for Students in Computer Science and Information Systems. S.-V. L. L. 2008, Ed. Second., pp. 1–162. London.

[12] Nair, D. G., P, B. V, & Kumar, G. S. (2015). An Effective Private Data storage and Retrieval System using Secret sharing scheme based on Secure Multi-party Computation. *IEEE*, 210–214. Chaotic Dynamics; Atmospheric and Oceanic Physics. doi:10.1063/1.4908174

[13] Miruthuladevi, N., Priyanga, P., Ramya, R., & Shobana, M. (2014). ABE Enforced Triple-DES with Outsourced Decryption in Cloud. *International Journal of Computer Science & Engineering Technology (IJCSET)*, *5*(03), 214–218.

[14] Saranya, S., & Shankar, K. N. V. (2014). An Enhanced Attribute Based Encryption with Multi Parties Access in Cloud Area. *International Journal of Computer Science and Mobile Computing*, *3*(1), 585–590.

[15] Mouftah, H. T., & Kantarci, B. (2014). Accelerating Mobile- Cloud Computing: A Survey. In *Communication Infrastructures for Cloud Computing* (Vol. i, pp. 175–197).

[16] Qi, H., & Gani, A. (2012). Research on Mobile Cloud Computing : Review , Trend and Perspectives. In *IEEE Second International Conference on Digital Information and Communication Technology and its Applications* (pp. 195–202).